

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Артеменков Михаил Николаевич
Должность: Ректор
Дата подписания: 07.10.2026 12:25:58
Уникальный программный ключ:
a63b74166e8f4fc17bd1e0462725f79e393b8f9

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Смоленский государственный университет»

Кафедра прикладной математики и информатики

«Утверждаю»

Проректор по учебно-методической деятельности

_____ Ю.А. Устименко
«25» декабрь 2025 г.

**Рабочая программа дисциплины
Б1.О.06.14 Практикум по решению задач на ЭВМ**

Направление подготовки **44.03.05 Педагогическое образование (с двумя профилями подготовки)**

Направленность (профиль) **Математика, Физика и информатика**

Форма обучения: очная

Курс – 3, 4, 5

Семестр – 6, 7, 8, 9

Всего зачетных единиц – 8, часов – 288

Форма отчетности: зачет – 6, 7, 8 семестр, экзамен – 9 семестр.

Программу разработала: кандидат педагогических наук, доцент Киселева О.М.

Одобрена на заседании кафедры
«20» ноября 2025 г., протокол № 5

Заведующий кафедрой _____ С.В. Козлов

Смоленск
2025

1. Место дисциплины в структуре ОП

Дисциплина «Практикум по решению задач на ЭВМ» относится к дисциплинам обязательной части основной образовательной программы направления подготовки 44.03.05 Педагогическое образование (с двумя профилями подготовки), направленность (профиль) образовательной программы: Математика, Физика и информатика. При изучении данной дисциплины необходимы компетенции студентов, сформированные при изучении дисциплин «Алгоритмы и структуры данных», «Программирование» и прохождения практики «Языки и методы программирования» и др.

Будущему учителю информатики важно глубоко разбираться во множестве современных видов и методов решения задач на ЭВМ. Поэтому компетенции, сформированные при изучении дисциплины, необходимы для последующего изучения профессиональных курсов, написания выпускной квалификационной работы бакалавра и его дальнейшей профессиональной деятельности.

2. Планируемые результаты обучения по дисциплине

Компетенция	Индикаторы достижения
ОПК-8 - Способен осуществлять педагогическую деятельность на основе специальных научных знаний	Знать: объект, предмет, основные категории, принципы, закономерности, структуру педагогической науки; сущность, структуру, динамику целостного педагогического процесса; состояние и тенденции развития отечественных и международных педагогических и психологических исследований; методологию педагогического исследования; особенности, логику, закономерности, формы, методы и средства процесса обучения и воспитания; основы психологии личности, основные теоретические подходы к пониманию феномена личности; познавательные процессы, их свойства, закономерности и роль в интеллектуальной и творческой деятельности; общетеоретические основы методики преподавания предмета в объеме, необходимом для осуществления педагогической деятельности; строение и функции организма, основные закономерности развития человека; общие закономерности и возрастные особенности функционирования основных систем организма учащихся; гигиенические требования к организации образовательного процесса и гигиену учебного процесса; инструментальные средства информационных технологий. Уметь: применять теоретические знания в решении педагогических задач; планировать, проектировать и осуществлять педагогический процесс в различных типах образовательных учреждений; определять структуру и методологию проведения педагогического исследования; адекватно целям выстраивать учебный и воспитательный процесс, выбирая соответствующие формы, методы и средства его осуществления; использовать в педагогической деятельности и межличностном взаимодействии современные достижения психологической науки; учитывать возрастные физиологические особенности учащихся в педагогическом процессе; использовать информационные технологии для решения профессиональных задач. Владеть: категориальным аппаратом педагогической науки; навыками решения педагогических задач; способами планирования и осуществления образовательного процесса; способами проведения педагогического эксперимента; формами и методами осуществления учебной и воспитательной работы; приемами и методами психодиагностики личности, изучения особенностей профессиональной деятельности; навыками организации педагогической деятельности с позиций сохранения здоровья;

	методами профилактики нарушений физического развития и повышения адаптационных резервов организма; методами оказания первой доврачебной помощи; методами применения информационно-коммуникационных технологий в образовательном процессе.
ПК-6 - Способен использовать научные знания в предметной области (информатика) в процессе формирования предметной компетенции обучающихся в рамках реализации основной общеобразовательной программы	Знать: назначение, структуру и содержание курса информатики, современное состояние и перспективы развития информатики как учебной дисциплины, ее место и роль, фундаментальное ядро современного школьного курса информатики, принципы построения методической системы обучения информатике, ее основных компонентов. Уметь: анализировать цели и содержание школьного курса информатики, проектировать образовательный процесс, использовать дидактический потенциал средств информационных технологий в реализации образовательного процесса по курсу информатики; Владеть: основными видами профессиональной деятельности учителя информатики, профессиональными навыками реализации методики обучения основным разделам курса информатики, современными информационно-коммуникационными средствами для эффективного осуществления профессиональной деятельности.

3. Содержание дисциплины

6 семестр

Основы теории проектирования баз данных. Определение и назначение баз данных. Системы управления базами данных — СУБД. Области применения баз данных. Информационная модель данных, ее состав (концептуальная, логическая и физическая модели). Три типа логических моделей баз данных: иерархическая, сетевая, реляционная. Типы взаимосвязей в модели: «один к одному», «один ко многим», «многие ко многим». Обеспечение непротиворечивости и целостности данных в базе. Нормализация баз данных. Этапы проектирования баз данных.

Использование СУБД ACCESS для создания баз данных. Основные характеристики и возможности СУБД Access. Основные компоненты СУБД Access. Типы данных СУБД Access. Создание новой базы данных. Создание таблиц в СУБД Access. Схема данных в Access. Модификация структуры базы данных. Обработка данных в базе. Запросы в СУБД Access. Основы конструирования запросов. Условия отбора записей, сортировка и фильтрация данных. Изменение данных в БД средствами запроса. Формы — диалоговый графический интерфейс для работы пользователя с базой данных. Основы создания формы. Элементы управления. Технология загрузки, просмотра и корректировки данных базы с использованием форм. Разработка многотабличных форм. Разработка отчетов.

7 семестр

Среда редактора VB. Модули в Word, Excel, PowerPoint. Редактор Visual Basic. Редактирование простых макросов.

Структура программы VB. Типы данных VBA. Объявление простых переменных. Объявление констант. Массивы. Записи и типы, указанные пользователем. Выражения. Правила построения выражений в VBA. Работа с числовыми данными. Работа со строками. Работа с датами и временем. Функции преобразования данных. Автоматическое преобразование данных в VB. Оператор присваивания. Логические операторы. Управляющие операторы. Работа с файлами и папками. Классификация процедур. Синтаксис процедур и функций.

Разработка приложений. Работа с объектами Excel. Формы пользователя. Элементы управления. Использование панели элементов (Tool-box). Процесс разработки приложения с диалоговой формой. Отладка VB-кода. Поиск и устранение ошибок. Работа с объектами Excel.

Разработка приложений. Работа с объектами Word. Работа со связанными и внедренными объектами. Работа с объектами Word.

Доступ к базам данным из MS Office. Доступ к базам данным из VBA-кода. Использование библиотеки ADO.NET. Представления и хранимые процедуры. Печать отчетов: Word и Excel с использованием VBA-кода.

8 семестр

Комбинаторные и рекурсивные алгоритмы. Различные способы представлений в ЭВМ конечных последовательностей и операции над ними. Длина путей. [Комбинаторика разбиений](#). Разные статистики. Деревья и перестановки из n элементов. Число сочетаний C_m^n . Задачи на разбиение чисел. Комбинаторные задачи теории информации. Множества. Решето Эратосфена. [Рекуррентные соотношения](#). Размещения без повторений. Перестановки. Сочетания. Задача: «Затруднение мажордома». [Алгоритмы рекуррентных соотношений](#). Ханойская башня. Разновидности связанных списков. Стеки и очередь. Задачи. Деревья. Представления. Прохождения.

Алгоритмы поиска. Последовательный поиск. Поиск в неупорядоченных массивах. Поиск элемента с дополнительными условиями. Поиск в упорядоченных массивах. Логарифмический поиск в статических таблицах. Бинарный поиск. Оптимальные деревья бинарного поиска. Логарифмический поиск в динамических таблицах. Сбалансированные сильно ветвящиеся деревья.

Алгоритмы сортировки. Сортировка данных. Внутренняя сортировка. Вставка. Сортировка простыми вставками. Обменная сортировка. Сортировка выбором. Шейкерная и гномья сортировки. Распределяющая сортировка. Внешняя сортировка. Частичная сортировка (выбор). Частичная сортировка (слияние).

Алгоритмы на графах. Поиск в глубину. Алгоритм Дейкстры нахождения кратчайшего пути. Алгоритм Флойда нахождения кратчайших путей между парами вершин. Автоматическое построение лабиринтов. Бинарное дерево. Задача о восьми ферзях. Задача о назначениях (задачи выбора). Алгоритмы для решения задач на графах. Процедура поиска в ширину. BFS-дерево и вычисление расстояний. Процедура поиска в глубину. DFS-дерево. Глубинная нумерация. Построение каркаса. Шарниры. [Эйлеровы и гамильтоновы циклы](#) Построение эйлерова цикла. Независимые множества, клики, вершинные покрытия. Стратегия перебора для задачи о независимом множестве. Эвристики для задачи о независимом множестве. Перебор максимальных независимых множеств. [Раскраски](#). Раскраска вершин. Переборный алгоритм для раскраски. Раскраска ребер. [Рационализация переборных алгоритмов](#) Рационализация поиска наибольшего независимого множества. Хордалльные графы. Рационализация алгоритма для задачи о раскраске вершин. [Оптимальные каркасы](#) Задача об оптимальном каркасе. Алгоритм Прима. Алгоритм Краскала. [Жадные алгоритмы и матроиды](#). Кратчайшие пути, геодезическое дерево и алгоритм Дейкстры.

Задачи на клеточной карте. Клеточная карта. Задача о строительстве порта. Задача о поиске кратчайшего пути. Задача о нахождении периметра острова. Задача о подсчёте числа островов. Раскрашивание островов. Задача о нахождении площади острова. Задача о поиске выхода из лабиринта.

Задачи на шахматной доске. Расстановка фигур на шахматной доске. Задача о ферзях.

9 семестр

Среда MIT App inventor. Интерфейс пользователя. Описание интерфейса пользователя. Режим “Дизайнер”. Экраны приложения. Режим “Блоки”. Функции режима “Блоки”. Загрузка и установка приложения на устройство. Загрузка .apk файла на мобильное устройство. Компоненты приложения. Разрешение экрана.

Практические приемы создания приложений. Кнопки. Приложения с несколькими экранами. Обмен данными между экранами. Списки. Рисование. Анимация. Медиа. Общение. Сенсоры. Математические функции

Организация проектной деятельности. Совместная разработка приложений. Создание итогового проекта

4. Тематический план

№ п/п	Разделы и темы	Всего часов	Формы занятий	
			лабораторные занятия	самостоятельная работа
Семестр 6				
1	Основы теории проектирования баз данных	32	20	12
2	Использование СУБД ACCESS для создания баз данных	40	28	12
	Итого	72	48	24
Семестр 7				
1	Среда редактора VB	14	6	8
2	Структура программы VB	14	6	8
3	Разработка приложений. Работа с объектами Excel	16	6	10
4	Разработка приложений. Работа с объектами Word	14	6	8
5	Доступ к базам данным из MS Office	14	6	8
	Итого	72	30	42
Семестр 8				
1	Комбинаторные и рекурсивные алгоритмы	12	4	8
2	Алгоритмы поиска	12	8	4
3	Алгоритмы сортировки	12	8	4
4	Алгоритмы на графах	12	4	8
5	Задачи на клеточной карте	12	8	4
6	Задачи на шахматной доске	12	8	4
	Итого	72	40	32
Семестр 9				
1	Среда MIT App inventor	6	4	2
2	Практические приемы создания приложений	22	20	2
3	Организация проектной деятельности	17	8	9
	Экзамен	27	-	27
	Итого	72	32	13
ИТОГО		288	150	111+27

5. Виды образовательной деятельности

Занятия семинарского типа

Лабораторные занятия

Задания для лабораторных работ размещены в системе дистанционного обучения Смоленского государственного университета в категории Физико-математический факультет.

6 семестр

Лабораторная работа 1. Создание таблиц данных в режиме Конструктора с помощью Мастера и Мастера подстановок

1. Понятие реляционной модели данных.
2. Понятие таблицы, создание таблиц с помощью конструктора и в режиме таблицы.
3. Типы данных и их представление.
4. Изменение типа полей и реорганизация таблиц.
5. Создание и редактирование таблиц базы данных с помощью мастера создания таблиц и с помощью мастера подстановок. Изменение внешнего вида таблицы.
6. Выбор используемых шрифтов Изменение ширины столбца и высоты строк.

Лабораторная работа 2. Фильтрация данных, основные виды фильтров

1. Фильтр, понятие фильтра.
2. Способы фильтрации, фильтр по форме, расширенный фильтр, фильтр по выделенному.
3. Поле Фильтр. Использование команды Исключить выделенное.

Лабораторная работа 3. Установка связей между таблицами

1. Сложные, многофункциональные таблицы данных.
2. Связи между таблицами, основные виды связей.
3. Обеспечение целостности данных в базе.
4. Редактирование и удаление связей между таблицами.
5. Отображение связанных записей в режиме таблицы.

Лабораторная работа 4. Создание простых запросов в режиме конструктора и с помощью мастера запросов

1. Поиск информации по базе данных.
2. Основные типы запросов. Создание запросов на выборку с помощью конструктора и мастера запросов.
3. Запросы на изменение: обновление, добавление, удаление и создание новой таблицы.

Лабораторная работа 5. Создание одно- и многопараметрических запросов

1. Запросы с параметрами.
2. Виды запросов с параметрами.
3. Методы их построения.

Лабораторная работа 6. Создание запросов с использованием вычисляемых полей

1. Формат вычисляемого поля.
2. Условия отбора для вычисляемых полей.
3. Использование построителя выражений.
4. Использование параметра в вычисляемых полях.

Лабораторная работа 7,8. Создание простых форм с помощью мастера и их усложнение с помощью конструктора

1. Создание форм с помощью мастера.
2. Конструктор форм.
3. Использование элементов управления.
4. Модифицирование формы. Перемещение элементов управления.
5. Независимое перемещение элементов управления и их надписей.
6. Изменение размеров элементов управления и надписей. Разделы формы. Создание списка или поля со списком.
7. Создание группы переключателей.

8. Создание элемента управления Кнопка.
9. Форматирование текстовых элементов формы.

Лабораторная работа 9,10. Создание простых отчетов с помощью мастера отчетов и их редактирование в режиме конструктора

1. Использование мастера отчетов.
2. Общая структура отчета. Настройка отчета в режиме конструктора.
3. Операции с элементами управления отчета.
4. Нумерация страниц и указание даты.
5. Форматирование отчета. Обрамление, разделение страниц и вставка рисунков.
6. Использование линий и прямоугольников. Разделители страниц.
7. Добавление рисунков в отчеты. Группирование и сортировка записей в отчетах.
8. Добавление вычисляемых элементов управления Использование мастера диаграмм.
9. Построение исходного запроса для диаграммы. Создание диаграммы с помощью мастера.

Лабораторная работа 11,12. Использование СУБД ACCESS для создания баз данных

Создайте новую базу данных под именем **Учебная (номер группы).mdb**. Для этого при первом запуске Access в появившемся окне **Создание файла** выбрать **Новая база данных**.

Примечание. Если вместо mdb ввести другое расширение, то Access сохранит файл базы данных, но в дальнейшем не будет отображать его в списке существующих баз данных. По умолчанию система обращается только к файлам с расширением mdb.

После создания базы данных появляется панель инструментов (рис. 1), позволяющая быстро создавать таблицы, запросы, формы, отчеты и т.д.

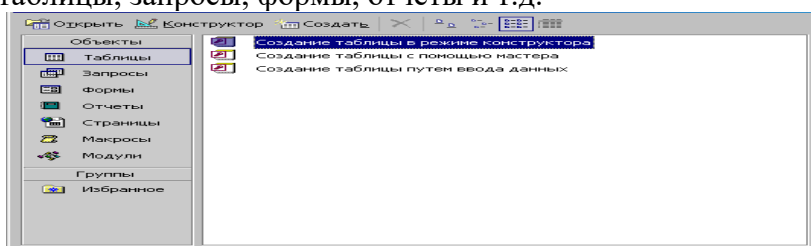


Рис. 1. Панель инструментов

В объектах панели инструментов выбираем **Таблицы** и в левой части **Создание таблицы в режиме конструктора**. В режиме **Конструктор** создайте таблицу данных, структура которой представлена в таблице 1.

Таблица 1

Имя поля	Тип данных	Описание
Код товара	Счетчик	
Марка автомобиля	Текстовый	
Фирма производитель	Текстовый	
Год выпуска	Дата/время	Месяц и год выпуска
Цена	Денежный	Стоимость единицы товара

Введите дополнительные свойства полей (см. табл. 2).

Таблица 2

Имя поля	Размер поля	Формат поля
Марка автомобиля	30	
Фирма производитель	40	
Год выпуска		Краткий формат даты
Цена		Денежный

Установите для поля **Код товара** признак ключевое и сохраните таблицу под именем **Товар**.

Откройте таблицу **Товар** и заполните ее данными, представленными в таблице 3.

Таблица 3

Марка автомобиля	Фирма производитель	Год выпуска	Цена
Toyota-avensis	Тойота	20.09.99	390 450
Toyota Solara	Тойота	10.04.05	513 000
Volvo-V40	Ltd. Вольво	03.10.98	342 000
Toyota-Nadya	Тойота	05.07.00	420 000
Renault-Laguna	Рено	04.08.01	280 000
Renault-Europa	Рено	12.09.03	320 000
VW-Golf-IV	Концерн VW	09.09.04	385 000
VW-Golf-IV	Концерн VW	01.09.05	450 000
VW-Golf-III	Концерн VW	14.02.93	175 500
VW-Golf-III	Концерн VW	05.07.00	210 000
Volvo-S90	Ltd. Вольво	22.11.96	280 000
BA3-21074	BA3	01.01.10	130 000
BA3-2105	BA3	12.01.10	125 000
BA3-2109	BA3	04.02.10	175 000
BA3-21099	BA3	07.06.10	190 000
BA3-2110	BA3	07.06.10	210 000
BA3-2115	BA3	17.09.10	225 000
BA3-2115	BA3	10.07.10	223 000
BA3-2105	BA3	11.09.04	105 000
BA3-2106	BA3	23.08.05	128 000

С помощью мастера создайте таблицу **Покупатели**. Для этого выберите категорию **Деловые** и образец таблицы **Клиенты**. В новую таблицу добавьте из образцов полей следующие поля: Код Клиента, Название Компании, Город и Номер телефона. Код Клиента переименуйте в Код покупателя, так как это показано на рисунке 2.

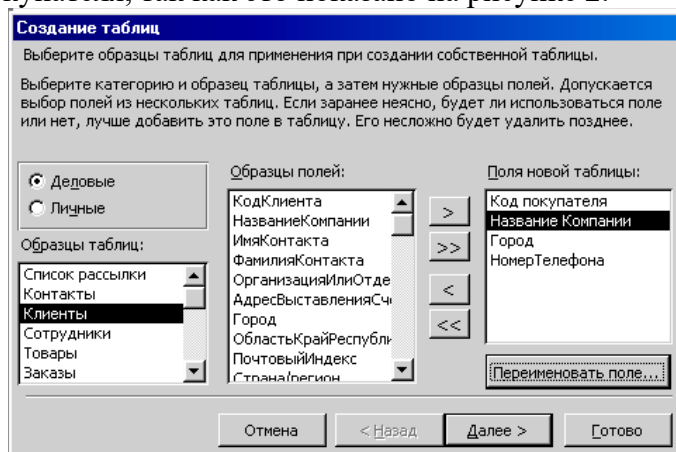


Рис. 2. Создание таблицы Покупатели

Присвойте имя таблице **Покупатели**, самостоятельно задайте полю **Код покупателя** параметр ключевое. Таблицу **Покупатели** не связывать с таблицей **Товар**.

Откройте таблицу **Покупатели** и заполните ее данными, представленными в таблице 4.

Таблица 4

Название Компании	Город	Номер Телефона
Генезис	Москва	308-40-08
Астра	Тула	67-56-67

Шелтон	Воронеж	89-45-47
Мартод	Новосибирск	45-78-23
Буревестник	Владимир	23-89-45
Аритор	Рязань	54-66-78
Гранд	Москва	456-76-45
Мосавто	Москва	920-33-55

С помощью кнопок сортировки панели инструментов выполните сортировку по возрастанию названий компаний.

Примечание. Нельзя сортировать гиперссылки, МЕМО поля и объекты OLE.

В режиме **Конструктор** создайте еще одну таблицу данных, структура которой представлена в таблице 5.

Таблица 5

Имя поля	Тип данных	Описание
Код продажи	Счетчик	
Код покупателя	Числовой	
Код товара	Числовой	
Количество	Числовой	
Дата продажи	Дата/время	

Для поля **Дата продажи** установите формат поля – краткий формат даты, полю **Код продажи** задайте параметр ключевое и сохраните таблицу под именем **Продажа**. Откройте таблицу **Продажа** и заполните ее данными, представленными в таблице 6.

Таблица 6

Код покупателя	Код товара	Количество	Дата продажи
2	4	1	02.07.10
3	7	1	10.08.10
6	10	3	12.10.10
7	15	10	17.09.10
1	2	2	22.07.10
4	5	1	07.05.10
5	18	15	25.06.10
8	20	20	30.10.10

Создание таблицы в режиме таблицы

Единственным «показанием» к использованию именно этого метода создания таблицы является необходимость срочно ввести в табличную структуру некоторые данные. В режиме таблицы (напоминающем создание рабочего листа в приложении Microsoft Excel) программа Access сразу предоставляет в ваше распоряжение табличную структуру, состоящую из 10 полей и включающую 21 пустую запись. Чтобы создать таблицу в режиме таблицы, выполните следующее.

1. В окне базы данных щелкните на кнопке Таблицы, а затем дважды щелкните на ярлыке Создание таблицы путем ввода данных. В рабочем пространстве приложения Access появится новая таблица — Таблица 1 — в режиме таблицы (рис. 3).

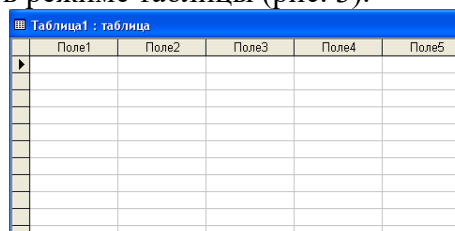


Рис. 3. Режим создания таблицы путем ввода данных

2. Укажите имена полей для новой таблицы. Для этого достаточно дважды щелкнуть на заголовке нужного столбца (Поле 1, Поле 2 и т.д.) и ввести имя поля.

3. Задав имена полей, можно вводить в таблицу данные.

4. Закончив ввод, сохраните новую таблицу, щелкнув на кнопке «Сохранить» панели инструментов (или выберите команду Файл → Сохранить) и в открывшемся окне введите имя новой таблицы и щелкните на кнопке ОК. При сохранении таблицы все пустые столбцы, которые не были переименованы, удаляются. По окончании ввода данных программа Access анализирует содержимое полей таблицы и сама задает для них тип данных, который более всего им соответствует. Однако рано или поздно вам все равно придется перейти в режим конструктора, чтобы уточнить типы полей (если Access их неверно истолкует), задать требуемые свойства полей, определить первичный ключ и выполнить другие важные операции, касающиеся структуры таблицы.

Создайте таблицу «Товар на сервисном ремонте» с помощью режима Создание таблицы путем ввода данных. Введите наименования полей: Код операции, Наименование товара, Фамилия клиента, Состояние, Дата приемки, Срок ремонта.

Заполните полученную таблицу данными, представленными в таблице 7.

Таблица 7

Код	Наименование товара	Фамилия клиента	Состояние	Дата приемки	Срок ремонта
1	BA3-2107	Иванов	В ремонте	12.07.10	20.07.10
2	BA3-2115	Сидоров	Диагностика	02.05.10	15.07.10
3	Toyota Solara	Новикова	Ждет деталей	12.04.10	12.05.10
4	VW-Golf-IV	Ушаков	Ждет деталей	23.04.10	04.05.10
5	Renault-Laguna	Иванов	В ремонте	20.05.10	20.06.10
6	BA3-2110	Воронов	В ремонте	18.04.10	19.05.10
7	BA3-2115	Смирнов	В ремонте	07.05.10	10.06.10
8	BA3-2126	Васильев	Диагностика	30.04.10	02.06.10
9	BA3-2107	Антонов	Диагностика	12.05.10	08.06.10
10	BA3-2114	Румянцев	Ждет деталей	14.05.10	04.06.10

Сохраните созданную таблицу под соответствующим названием, после чего откройте ее в режиме конструктора, удалите лишние столбцы, полю Код операции присвойте параметр ключевое поле. Будьте очень внимательны и применительно ко всем «структурным» операциям придерживайтесь золотого правила: семь раз отмерь... Чтобы уверенно чувствовать себя, не поленитесь создать резервную копию изменяемой таблицы. Сохраните внесенные изменения.

Примечание. Изменение типов полей может повлечь за собой потерю данных. Программа Microsoft Access предупреждает об этом в специальном сообщении прежде чем сохранить внесенные изменения. На данном этапе можно отказаться от изменений структуры таблицы, щелкнув на кнопке Отмена диалогового окна. Если вы уверены в своем выборе, щелкните на кнопке ОК. В приложении Access нельзя преобразовать данные какого-либо другого типа (даже числового) к типу Счетчик. Поля типа Счетчик - это служебные поля, которые хранят номера записей и не доступны для привычной операции ввода данных в ячейку поля. Поля, тип которых — *Поле объекта OLE*, не допускают никаких преобразований. В тип Текстовый могут быть преобразованы, например, данные типа Счетчик, Числовой или Денежный. Числовые и денежные данные после этой операции будут отображаться без знаков валют и разделителей, но дробные десятичные знаки у этих данных сохраняются.

Использование Мастера подстановок

Сделать более простым ввод значений в поле позволяет операция подстановки. Используя эту операцию, можно выбирать значения поля из списка. Список значений может быть как фиксированным, так и содержаться в таблице или запросе. Сформировать столбец подстановок для поля помогает Мастер подстановок.

Создайте таблицу «Продажа по собственной фирме». Выберите режим «Создание таблицы в режиме конструктора».

Структура таблицы представлена в таблице 8.

Таблица 8

Имя поля	Тип данных	Описание
Код сотрудника	Счетчик	
Фамилия	Текстовый	
Должность	Мастер подстановок	

В появившемся диалоговом окне выберите способ ввода данных «Будет введен фиксированный набор значений». С клавиатуры заполните Столбец 1, как показано на рисунке 4.

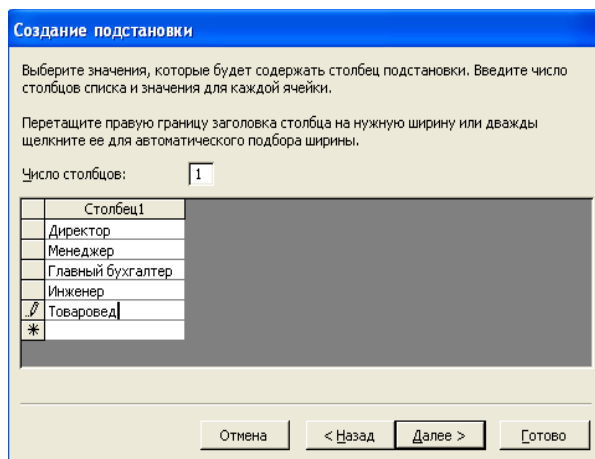


Рис. 4. Режим подстановки в Мастере подстановок

Потом нажмите «Далее» и в итоговом диалоговом окне «Готово».

Кроме этого, данные можно в Мастере подстановок вводить и из другой таблицы или запроса. Добавьте в структуру строку, как показано в таблице 9.

Таблица 9

Имя поля	Тип данных	Описание
Код сотрудника	Счетчик	
Фамилия	Текстовый	
Должность	Мастер подстановок	
Марка купленной машины	Мастер подстановок	

В окне Мастера подстановок выберите «Объект «столбец подстановки» будет использовать значения из таблицы или запроса».

В следующем окне выберите таблицу Товар. Нажмите Далее и выберите поле Марка автомобиля, нажмите Далее и Готово. Сохраните таблицу под именем Продажа сотрудникам фирмы.

Откройте таблицу и заполните ее следующими данными.

Фамилия	Должность	Марка купленной машины
Иванов	Менеджер	Volvo-V40
Петров	Директор	Toyota-Solara
Соколова	Главный бухгалтер	VW-Golf-IV
Сидоров	Товаровед	BA3-2115
Пеньков	Инженер	BA3-2106

Самостоятельная работа

1. Из каких этапов складывается проектирование базы данных?
2. Зачем перед созданием базы данных нужно определить структуру таблиц?

3. Какими принципами нужно руководствоваться при создании таблиц?
4. Какие объекты базы данных предусмотрены в Access?
5. Какие режимы предназначены в Access для работы с таблицами?
6. Как вызывается режим конструктора таблиц?
7. Каковы операции задания структуры таблицы с помощью конструктора?
8. Какие типы данных предусмотрены в Access?
9. В чем состоит особенность поля типа Счетчик?
10. Как и зачем вводится в таблицы ключевое поле?
11. Какие способы ввода данных в таблицы вам известны?
12. Как перемещаться между различными ячейками таблицы?

Лабораторная работа 13,14. Использование СУБД ACCESS для создания баз данных

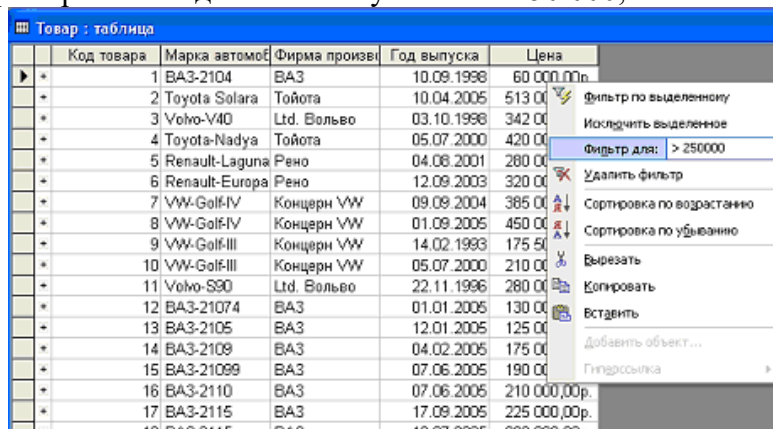
В качестве примера выполним в таблице Товар отбор записей о товарах, цена которых превышает 250 000 рублей. Чтобы использовать для этих целей поле Фильтр для, выполните следующее.

Откройте таблицу **Товар**, в которой необходимо выполнить поиск в режиме таблицы, и щелкните правой кнопкой мыши на нужном поле (например, поле Цена). Появится контекстное меню, в котором содержится поле Фильтр для.

Введите в поле Фильтр для требуемый критерий фильтрации.

Примечание. Таким критерием может быть как конкретное значение, так и некоторое выражение.

Для нашего примера необходимо ввести условие $> 250\,000$, как показано на рисунке 5.



	Код товара	Марка автомобиля	Фирма-производитель	Год выпуска	Цена
1	BA3-2104	BA3		10.09.1998	60 000,00р.
2	Toyota Solara	Тойота		10.04.2005	513 000,00р.
3	Volvo-V40	Ltd. Вольво		03.10.1998	342 000,00р.
4	Toyota-Nadysa	Тойота		05.07.2000	420 000,00р.
5	Renault-Laguna	Рено		04.08.2001	280 000,00р.
6	Renault-Europa	Рено		12.09.2003	320 000,00р.
7	VW-Golf-IV	Концерн VW		09.09.2004	385 000,00р.
8	VW-Golf-IV	Концерн VW		01.09.2005	450 000,00р.
9	VW-Golf-III	Концерн VW		14.02.1993	175 500,00р.
10	VW-Golf-III	Концерн VW		05.07.2000	210 000,00р.
11	Volvo-S80	Ltd. Вольво		22.11.1996	280 000,00р.
12	BA3-21074	BA3		01.01.2005	130 000,00р.
13	BA3-2105	BA3		12.01.2005	125 000,00р.
14	BA3-2109	BA3		04.02.2005	175 000,00р.
15	BA3-21099	BA3		07.06.2005	190 000,00р.
16	BA3-2110	BA3		07.06.2005	210 000,00р.
17	BA3-2115	BA3		17.09.2005	225 000,00р.
18	BA3-2115	BA3		10.07.2005	223 000,00р.

Рис. 5. Ввод условия поиска для поля Фильтр для

Нажмите клавишу <Enter>, чтобы выбрать команду Фильтр для. Теперь в таблице будут отображены только те записи, которые удовлетворяют заданному условию отбора.

Чтобы удалить Фильтр, щелкните на кнопке Удалить фильтр (она активизирована на панели инструментов Таблица в режиме таблицы) или выберите команду главного меню Записи→Удалить фильтр.

Еще один способ фильтрации данных, команда Фильтр по выделенному, находит в таблице те записи, которые соответствуют некоторому выделенному фрагменту данных таблицы. Поэтому первое, что необходимо сделать, используя Фильтр по выделенному, — это найти в таблице запись, в поле которой содержится значение, принимаемое в качестве критерия отбора.

Итак, чтобы применить фильтр по выделенному, выполните следующее.

В таблице **Товар** щелкните на столбце **Фирма-производитель**, найдите ячейку со значением **Ltd. Вольво**. Выделите в ячейке данные, которые должны присутствовать в полях результирующего набора записей. Щелкните на кнопке Фильтр по выделенному панели инструментов или выберите команду главного меню Записи→Фильтр→Фильтр по выделенному.

Записи исходной таблицы будут отфильтрованы, и в окне таблицы появятся только те из них, которые имеют в указанном поле значение, совпадающее с выделенным образцом. Кроме

того, кнопка Применение фильтра теперь активизирована и имеет другое название — Удалить фильтр.

Программа Access предусматривает, кроме того, возможность просмотреть те записи таблицы, которые при применении команды Фильтр по выделенному остаются «за кадром». То есть результатом использования фильтра в этом случае становится отображение записей, которые, наоборот, не удовлетворяют условию отбора. Чтобы отфильтровать записи по такой схеме, следует выполнить последовательность действий, как и при предыдущей фильтрации, но на последнем шаге вместо команды Фильтр по выделенному выбрать команду Записи→Фильтр→Исключить выделенное. Командой Исключить выделенное можно воспользоваться, когда в качестве критерия отбора выделяются данные из одной ячейки.

Например, чтобы просмотреть все записи, за исключением тех, которые относятся к товарам фирмы ВАЗ, следует найти в поле **Фирма производитель** таблицы Товар запись со значением **ВАЗ**, щелкнуть на этом значении правой кнопкой мыши и выбрать из открывшегося меню команду Исключить выделенное.

Двигаясь от простого к более сложному, рассмотрим следующий способ фильтрации, доступный в программе Microsoft Access. В отличие от фильтра по выделенному, так называемый *фильтр по форме*, или *обычный фильтр*, позволяет выполнить фильтрацию уже не по одному, а сразу по нескольким критериям, причем с использованием логического оператора Or (ИЛИ) для каждой группы заданных значений.

Рассмотрим работу данного фильтра. Для этого откройте таблицу **Товар**. Щелкните на кнопке Изменить фильтр панели инструментов или выберите команду Записи→Фильтр→Изменить фильтр.

На экране откроется окно фильтра по форме, которое позволяет указать условия отбора записей при фильтрации (рис. 6). При открытии окна фильтра по форме под главным меню появляется новая панель инструментов Фильтр, в строке состояния программы Access выводится надпись Режим формы, а кнопка раскрывающегося списка находится в том поле, которое было активным непосредственно перед открытием этого диалогового окна.

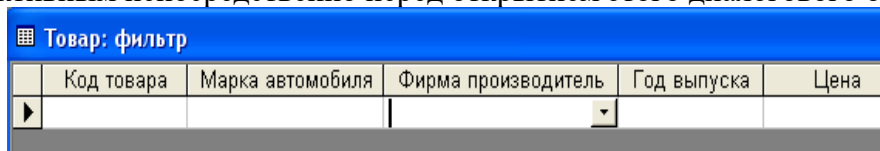


Рис. 6. Панель фильтра по форме

Чтобы задать условие отбора, перейдите к нужному полю и щелкните на нем. В правой части поля появится знакомая кнопка раскрывающегося списка. Щелкните на этой кнопке и выберите из списка искомое значение.

Для примера в поле **Фирма производитель** выберите **ВАЗ**. Программа Access автоматически заключает выбранное значение в кавычки.

Примечание. В диалоговом окне фильтра по форме условия отбора можно также вводить с клавиатуры или указывать с помощью выражений.

В поле **Цена** введите **> 200 000**. Указав все условия отбора в окне фильтра по форме, щелкните на кнопке Применение фильтра, расположенной на панели инструментов Фильтр, или выберите команду Фильтр → Применить фильтр.

В окне таблицы появятся записи, удовлетворяющие условиям отбора.

Созданный фильтр сохранить в виде запроса. Чтобы сделать это, следуйте таким инструкциям.

1. Перейдите в режим отображения фильтра в окне фильтра по форме.
2. Щелкните на кнопке Сохранить как запрос панели инструментов Фильтр или выберите команду главного меню Файл → Сохранить как запрос. Откроется небольшое диалоговое окно сохранения.
3. Введите имя **Фильтр по форме**, под которым будет сохранен новый запрос, и щелкните на кнопке ОК. В дальнейшем воспользоваться им можно будет, открыв вкладку Запросы в окне базы данных.

Последняя и самая мощная по своим возможностям команда фильтрации — это Расширенный фильтр, который вполне можно рассматривать как некое промежуточное звено между простыми операциями поиска и упорядочения данных в таблицах и настоящими запросами. Подробное знакомство с последовательностью действий при использовании команды Расширенный фильтр может дать первоначальное представление о том, как в программе Access создаются подлинные запросы. Тем не менее при всех своих преимуществах расширенный фильтр остается фильтром и не позволяет, например, производить анализ данных одновременно из нескольких таблиц, указывать поля, которые будут отображаться на экране для результирующего набора записей, а также выполнять вычисления на основе значений полей.

Для построения расширенного фильтра используется специальный инструмент — конструктор, доступный с помощью команды главного меню Записи → Фильтр → Расширенный фильтр. Окно конструктора выглядит аналогично представленному на рисунке 7.

Как несложно догадаться, фильтр применяется к таблице Товар. В верхней области окна конструктора расположен *список полей*, в котором перечислены все поля текущей таблицы. Нижняя панель окна — это бланк, в который вносятся условия отбора записей при применении расширенного фильтра.

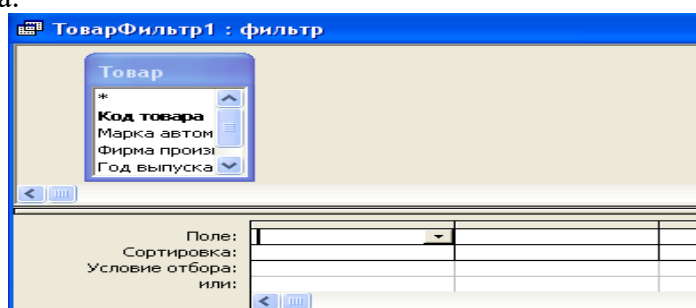


Рис. 7. Окно конструктора расширенного фильтра

В качестве примера создадим расширенный фильтр, который формировал бы список товаров стоимостью более 200 000 рублей, произведенных в период с 12 января по 17 сентября 2010 года. Чтобы построить простейший запрос (или расширенный фильтр), выполните следующее.

1. Откройте таблицу **Товар** в режиме таблицы.
2. Выберите команду главного меню Записи → Фильтр → Расширенный фильтр. На экране появится диалоговое окно конструктора расширенного фильтра. (Если в бланке фильтра уже указаны какие-либо условия выбора (сохраненные программой Access для ранее применявшегося фильтра), удалите их, выбрав команду главного меню Правка → Очистить бланк.)
3. Щелкните в первой ячейке строки Поле нижней панели окна. В правой части выбранного поля появится кнопка раскрывающегося списка. Щелкните на этой кнопке и выберите из списка нужное имя поля, по которому будет задано условие отбора. Для нашего примера следует выбрать поле **Цена**. Программа Access поместит название выбранного поля в текстовое поле Поле.

Примечание. Название поля, по которому будет выполняться фильтрация, можно перетащить из списка полей таблицы, расположенного в верхней области окна конструктора, или дважды щелкнуть на имени поля в этом списке.

4. В поле Условия отбора введите > 200 000. Во втором столбце выберите имя поля **Год выпуска** и в поле Условия введите выражение >#12.01.10# And <#17.09.10#.
5. Когда все условия отбора заданы, щелкните на кнопке Применение фильтра, расположенной на панели инструментов Фильтр, или выберите команду Фильтр → Применить фильтр. В окне таблицы появится результат применения расширенного фильтра.

Сохраните фильтр в виде запроса под именем **Расширенный фильтр**.

Самостоятельная работа

1. Как осуществить сортировку записей по возрастанию (по убыванию)?
2. В чем состоит преимущество фильтрации перед сортировкой?

3. Как задать простой фильтр?
4. Как задается расширенный фильтр?
5. Как выполняется поиск нужной информации в базе данных?

Лабораторная работа 15,16. Использование СУБД ACCESS для создания баз данных

Установите связь между таблицами **Товар**, **Покупатели** и **Продажа**. Для этого зайдите в пункт меню **Сервис** и выберите пункт **Схема данных**. В диалоговом окне **Добавление таблицы** добавьте таблицы **Товар**, **Продажа** и **Покупатели**. Создайте связи между таблицами. Для этого выберите таблицу **Покупатели**, в которой общее поле **Код покупателя** играет роль первичного ключа. Щелкните на этом поле и, удерживая левую кнопку мыши, перетащите его на таблицу **Продажа**, поместив точно над тем полем, которое является общим. Переместив поле, отпустите кнопку мыши. Аналогично проделайте эту же операцию для таблиц **Товар** и **Продажа**, как показано на рисунке 8.

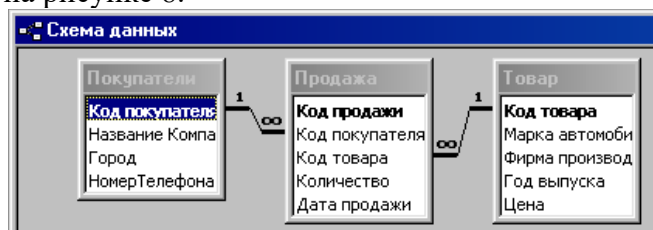


Рис. 8. Установка связей между таблицами

В появившемся диалоговом окне обеспечьте целостность данных, а также каскадное обновление и удаление связанных полей и записей.

Целостность данных — это система правил, регулирующих взаимодействие между связанными таблицами и обеспечивающих корректность хранимых в таких таблицах данных. Целостность данных подразумевает, что данные, введенные в общее поле двух связанных таблиц, должны совпадать. Как уже отмечалось, из двух связанных таблиц одна называется родительской, а вторая — дочерней, поэтому правильнее будет сказать, что данные, вводимые в поле внешнего ключа дочерней таблицы, должны совпадать с данными, хранимыми в поле первичного ключа родительской таблицы. В «фокусе» условий целостности всегда находятся ключевые поля связанных таблиц, поэтому при любых попытках их модификации программа Access автоматически проверяет их значения. Если такое изменение нарушает установленную между таблицами связь, выдается сообщение о нарушении целостности данных.

Примечание. Обязательными условиями поддержки целостности данных в приложении Access являются следующие: обе таблицы должны принадлежать одной базе данных; связанное поле родительской таблицы должно быть ключевым или иметь уникальный индекс; должно поддерживаться соответствие между типами связанных полей.

При установленном флажке опции **Обеспечение целостности данных** рядом с полем родительской таблицы на связи появится символ «1», а рядом с полем дочерней таблицы — символ перевернутой восьмерки.

Кроме того, при выборе опции **обеспечения целостности** в окне **Изменение связей** доступными становятся еще две опции, которые позволяют активизировать автоматическое выполнение каскадного удаления и каскадного обновления данных.

Каскадное обновление связанных полей. Когда флажок этой опции установлен, при любом изменении данных первичного ключа в родительской таблице автоматически будут обновляться соответствующие значения в поле связанной таблицы. Благодаря внесению необходимых изменений в связанную таблицу целостность данных не будет нарушена. Если эта опция не включена, изменить значение ключевого поля первичной таблицы не удастся.

Каскадное удаление связанных записей. Когда установлен флажок этой опции, при удалении записи в родительской таблице автоматически будут удаляться все соответствующие записи в дочерней связанной таблице.

Убедившись, что параметры заданы корректно, установите связь между таблицами, щелкнув на кнопке **Создать**. В окне **Схема данных** между двумя связываемыми полями появится линия объединения. Закройте окно **Схема данных** и сохраните связи.

После установления связей между таблицами вид каждой из родительских таблиц базы данных несколько изменился. Все дело в том, что в режиме таблицы программа Access позволяет просматривать не только записи открытой родительской таблицы, но и данные, хранящиеся в подчиненной ей таблице. Откройте, на-

пример, таблицу **Товар**. Слева от каждой записи в таблице теперь отображен значок «плюс» (+), щелкнув на котором можно просмотреть связанные данные из таблицы **Продажа** для текущей записи.

Самостоятельная работа

1. Какие отношения между таблицами вам известны?
2. Что такое главная и связанная таблицы в отношении «один ко многим»?
3. Каким образом создаются связи между таблицами?
4. Как будет выглядеть линия связи в окне **Схема данных**, если не установить переключатель **Обеспечение целостности данных**?
5. Что собой представляет макет связей?
6. Как удалить связь между таблицами?
7. Что такое ключевое поле и поле внешнего ключа?

Лабораторная работа 17,18. Использование СУБД ACCESS для создания баз данных

В режиме конструктора запросов создайте запрос на выборку из таблицы **Товар** с параметрами таблицы 9.

Таблица 9

Поле	Код товара	Марка автомобиля	Фирма производитель	Год выпуска	Цена
Имя таблицы	Товар	Товар	Товар	Товар	Товар
Сортировка					
Вывод на экран	√	√	√	√	√
Условие отбора			ВАЗ		

В результате формируется запрос на выборку автомобилей, которые произведены фирмой ВАЗ. Сохраните запрос с именем **Автомобили ВАЗ** и просмотрите результаты запроса в режиме таблицы.

С помощью **Мастера** создайте запрос на выборку из таблицы **Покупатели**. Укажите имя запроса **Покупатели Москва** и, выбрав **Изменить макет запроса** для поля **Город** укажите условие отбора **Москва**.

Примечание. Условия отбора для запросов могут задаться не только по одному полю, но по нескольким сразу, а также носить строгое и нестрогое значение.

В качестве примера создайте запрос из таблицы **Продажа**, который выбрал бы только те операции продажи, где количество проданных автомобилей находится в диапазоне от 0 до 7 штук. Для этого в режиме конструктора запросов создайте запрос с параметрами таблицы 10.

Таблица 10

Поле	Код продажи	Код покупателя	Код товара	Количество	Дата продажи
Имя таблицы	Продажа	Продажа	Продажа	Продажа	Продажа
Сортировка					
Вывод на экран	√	√	√	√	√
Условие отбора				Between 0 And 7	

Сохраните запрос под именем **Продажа объема не выше 7** и просмотрите результаты запроса в режиме таблицы.

Рассмотрим диалоговое окно Создать, которое позволяет создавать более сложные разновидности запросов. Например, перекрестные запросы, запросы на поиск повторяющихся записей, запросы на поиск записей без подчиненных записей.

Мастер запросов на поиск повторяющихся записей. По заданному значению поля (или полей) таблицы этот мастер создает запрос на выборку, который позволяет обнаружить в таблице одинаковые записи. Мастер предлагает указать, по каким полям следует выполнять поиск повторяющихся значений и какие поля необходимо включить в результирующую таблицу. Как правило, помощь такого мастера оказывается весьма кстати при создании уникального ключевого поля для уже существующей таблицы. Кроме того, с помощью мастера можно выяснить, например, кто из клиентов оформил более одной сделки, или сопоставить стоимость одного и того же товара у разных поставщиков.

На панели управления запросами нажмите кнопку Создать. В диалоговом окне Новый запрос (см. рис. 9) выберите **Повторяющиеся записи** и нажмите ОК.

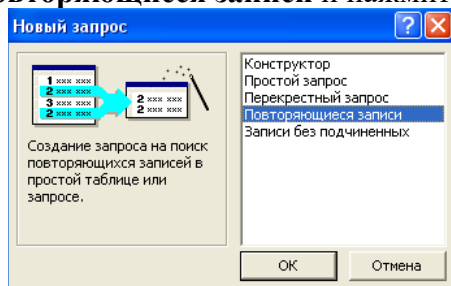


Рис. 9. Диалоговое окно Новый запрос

В появившемся диалоговом окне выберите таблицу **Товар на сервисном ремонте**. В качестве поля, содержащего повторяющиеся значения, укажите поле **Фамилия клиента**. В следующем окне переведите все доступные поля в раздел дополнительные, нажмите кнопку Далее и Готово. Результатом работы запроса будут две записи, показывающие, что у клиента Иванова два автомобиля находятся в ремонте.

Мастер запросов на поиск записей без подчиненных записей. Мастер просматривает данные двух связанных таблиц и отыскивает все записи родительской таблицы, для которой в подчиненной таблице отсутствуют соответствующие записи. Так, например, можно выяснить, какие товары не были еще включены в заказы или кто из клиентов пока воздерживается от покупок (поиск *записей-вдов*). Возможен и обратный подход, когда мастер помогает найти *записи-сироты* (например, в дочерней таблице имеются записи о так называемых «разовых» покупателях, не зарегистрированных в таблице постоянных клиентов, но уже оформивших покупки).

С помощью диалогового окна Новый запрос создайте запрос Записи без подчиненных. В роли родительской таблицы выберите таблицу **Покупатели**, а в качестве таблицы с подчиненными записями таблицу **Продажа** и нажмите Готово.

Запросы на изменение являются механизмами, автоматизирующими модификацию большого числа записей в таблицах. Основой для любого запроса на изменение служит обычный запрос на выборку, определяющий ту группу записей, над которыми будут выполнены некоторые действия. Поэтому при создании запроса на изменение сначала необходимо построить (в режиме конструктора или с помощью мастера) запрос на выборку, а затем изменить в режиме конструктора его тип.

Примечание. Так как запросы на изменение уже не просто осуществляют выборку определенной информации из таблиц, а модифицируют хранимые в них данные, привычная схема работы с запросом должна несколько измениться. Прежде, когда речь шла о «чистых» запросах на выборку, после создания запроса можно было смело щелкать на кнопке Запуск, не опасаясь за последствия такого шага. В случае с запросами на изменение ситуация кардинально отличается, так как команда Запуск сразу изменяет данные. Вернуть таблицу к исходному виду

невозможно. Единственный путь — обратиться к резервной копии таблицы, ставшей «жертвой» вашего эксперимента. Поэтому, чтобы предотвратить потерю важных данных, перед запуском любого запроса такого типа воспользуйтесь кнопкой Режим таблицы (одна из опций кнопки Вид панели инструментов окна конструктора), чтобы просмотреть те изменения, которые ожидают таблицу в результате выполнения запроса.

Чтобы проиллюстрировать, как может быть использован запрос на обновление, обратимся к простому примеру. Предположим, что необходимо повысить стоимость всех товаров фирмы ВАЗ на 10%.

Для этого создайте новый запрос на выборку, результирующими записями которого станут записи таблицы **Товар**, подлежащие изменению. Чтобы преобразовать запрос на выборку в запрос на обновление, выберите команду **Запрос → Обновление** или выберите из раскрывающегося списка кнопки Тип запроса панели инструментов пункт Обновление. Заголовок окна конструктора теперь сообщает о том, что запрос является запросом на обновление. Кроме того, в бланк запроса добавлена новая строка Обновление.

Введите для поля **Фирма производитель** условие отбора **ВАЗ**, а в ячейку строки **Обновление**, соответствующую столбцу **Цена**, выражение **[Товар]![Цена]*1,10** (см. рис. 10).

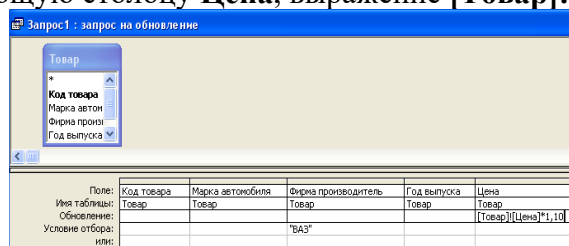


Рис. 10. Запрос на обновление

Нажмите на кнопке Запуск панели инструментов или выберите команду главного меню **Запрос → Запуск**. Программа Access сообщит о том, какое число записей будет добавлено в таблицу. Щелкните на кнопке Да, чтобы завершить выполнение запроса. Сохраните запрос под именем **Запрос на обновление**.

Запрос на создание таблицы. Запрос такого типа выполняется в два этапа: сначала осуществляется выборка необходимых записей, а затем на основе данных, полученных в результате этого запроса, создается новая таблица для текущей или какой-либо другой базы данных Microsoft Access. Одно из применений таких запросов — выделение некоторого подмножества записей, удовлетворяющих определенным условиям отбора, из одной или нескольких таблиц для их последующего анализа или обработки.

Например, необходимо выделить наиболее активных зарегистрированных покупателей, т.е. тех клиентов, которые за весь период обслуживания оформили больше всего заказов. Чтобы удобнее было работать с этими покупателями (например, сообщать им о предоставленных компанией определенных скидках), сведения о них должны быть размещены в отдельной таблице. Поэтому логично решить такую задачу, обратившись к запросу на создание таблицы.

Создайте в режиме конструктора запрос на таблицы **Покупатели** и **Продажа**, как показано на рисунке 11.

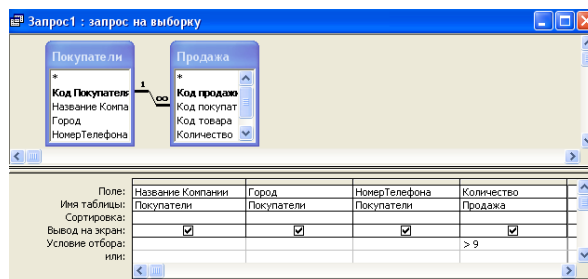


Рис. 11. Исходный запрос на выборку

В столбце **Количество** в ячейке **Условие отбора** введите выражение **>9**.

Выберите команду **Запрос → Создание таблицы** или воспользуйтесь для выбора типа запроса (на создание таблицы) кнопкой Тип запроса панели инструментов. На экране появится

диалоговое окно Создание таблицы. Введите в поле Имя таблицы название новой таблицы **Активные покупатели**, примите предложенную по умолчанию опцию в текущей базе данных и щелкните на кнопке ОК. В заголовке окна конструктора теперь указан другой тип запроса — на создание таблицы.

В режиме Конструктора запросов щелкните на кнопке **Запуск** панели инструментов или выберите команду главного меню **Запрос → Запуск**. В небольшом диалоговом окне программа Access сообщит о том, сколько записей копируется в новую таблицу, и запросит подтверждение на создание таблицы, затем щелкните на кнопке **Да**, чтобы завершить выполнение запроса.

Сохраните запрос под названием **Запрос на создание таблицы**.

Теперь, чтобы проверить полученные результаты, перейдите к окну базы данных, нажмите на вкладке **Таблицы** и откройте новую таблицу **Активные покупатели**.

Запрос на удаление. Как и следует из названия, главное и единственное назначение запросов такого типа — удалять из таблиц (или нескольких таблиц) записи, соответствующие определенному критерию. Именно записи, а не значения отдельных полей, для удаления которых можно воспользоваться запросом на обновление. Применение запросов на удаление обязывает быть предельно внимательным, поскольку удаление записей выполняется без возможности их последующего восстановления.

Создайте запрос на выборку для таблицы **Товар на сервисном ремонте** в режиме конструктора, добавьте в запрос все поля данной таблицы. В столбце **Наименование товара** в ячейке **Условие отбора** укажите **ВАЗ-2115**. Выберите команду **Запрос → Удаление** или выберите из раскрывающегося списка кнопки Тип запроса панели инструментов пункт **Удаление**. Запрос на выборку будет преобразован в запрос на удаление (о чем теперь сообщается в заголовке окна конструктора), а в бланке запроса вместо строки Сортировка появилась новая строка — Удаление.

Сохраните запрос под названием **Запрос на удаление**. Нажмите на данный запрос, программа Access сообщит о том, сколько записей будет удалено, и запросит подтверждение операции. Щелкните на кнопке **Да**.

Таков сценарий удаления записей из одной не связанной с другими таблицы. Однако на практике таблицы базы данных, как правило, связаны между собой, и наличие этих связей в схеме базы данных обязательно учитывается программой Access при попытке удалить записи одной из связанных таблиц. Напомним, что в случае, если между таблицами определено отношение «один ко многим» и в диалоговом окне Изменение связей установлен флажок опции каскадное удаление связанных записей, при удалении записи в главной таблице автоматически будут удалены все связанные записи подчиненной таблицы, даже если ее поля не фигурировали в бланке запроса на удаление.

Если флажок каскадного удаления не установлен, но для данного отношения между таблицами опция Обеспечение целостности данных активизирована, удалить записи из родительской таблицы не удастся до тех пор, пока не будут удалены записи из подчиненной таблицы. При попытке удаления записей программа Access выдает соответствующее предупреждение о наличии связанных записей в подчиненной таблице.

Самостоятельная работа

1. Чем запросы отличаются от фильтров?
2. Каково назначение запросов?
3. Какова процедура создания простого запроса в базе данных?
4. Как задать условие отбора в запросе?
5. Что такое запрос на создание таблицы?
6. Как создать запрос на обновление?
7. Какова процедура создания запроса на удаление?
8. Каков принцип работы Мастера запросов на поиск записей без подчиненных записей?
9. Каков принцип работы Мастера запросов на поиск повторяющихся записей?
10. Что такое перекрестный запрос?
11. Что такое запрос на добавление?

Лабораторная работа 19,20. Использование СУБД ACCESS для создания баз данных

Создайте однопараметрический запрос. Для этого в режиме конструктора добавьте таблицу **Покупатели**, выберите все поля таблицы и введите [Введите название компании] в **Условие отбора** для поля **Название Компании**, ввод квадратных скобок обязателен, именно они определяют, что запрос является параметрическим. Сохраните запрос с именем **Однопараметрический**. Запустите параметрический запрос, для этого щелкните по нему дважды левой клавишей мыши. На экране появится диалоговое окно, предлагающее ввести значение параметра (см. рис. 12).

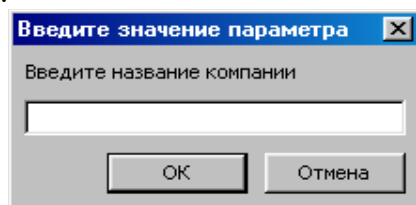


Рис. 12. Диалоговое окно ввода параметра

Введите название компании покупателя, например, Буревестник. После этого Access запускает запрос. Если параметр найден в таблице, то отобразятся данные, удовлетворяющие условию (см. рис. 13); в противном случае таблица запроса будет пустой.

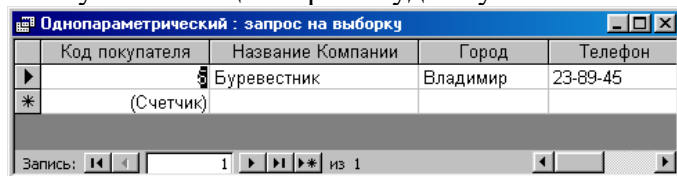


Рис. 13. Результат выполнения однопараметрического запроса

Создайте запрос, запрашивающий несколько параметров. В режиме конструктора добавьте все три таблицы и установите параметры, указанные в таблице 11.

Таблица 11

Поле	Марка автомобиля	Фирма производитель	Название компании	Количество	Год выпуска
Имя таблицы	Товар	Товар	Покупатели	Продажа	Товар
Сортировка					
Вывод на экран	√	√	√	√	√
Условие отбора	[введите марку автомобиля]		[введите название компании]	[количество проданных машин]	

Сохраните запрос под именем **Многопараметрический** и проверьте работоспособность запроса.

По умолчанию многопараметрический запрос выполняется слева направо, т.е. от Введите марку автомобиля до количество проданных машин. Этот порядок можно изменить. Для этого откройте запрос для редактирования в режиме конструктора, зайдите в пункт меню Запрос и выберите параметры, в появившемся диалоговом окне измените порядок вопросов, как показано на рисунке 14.

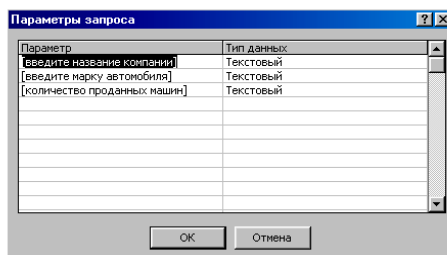


Рис. 14. Диалоговое окно параметров запроса

Примечание. Тексты приглашений ввода должны совпадать с таковыми в строке **Условие отбора** в окне конструктора запросов, иначе запрос будет работать некорректно.

Сохраните изменения запроса и проверьте новый порядок работы запроса.

Самостоятельная работа

1. Что такое запросы с параметрами?
2. Какова процедура создания однопараметрического запроса?
3. Какова процедура создания многопараметрического запроса?
4. Какие условия можно задавать в качестве параметров запроса?
5. Какова процедура изменения порядка работы многопараметрического запроса?

Лабораторная работа 21,22. Использование СУБД ACCESS для создания баз данных

Создайте запрос с использованием вычисляемого поля. В данном поле должна определяться сумма платежа, которую заплатит покупатель за купленный товар. Для этого запустите создание запроса в режиме конструктора, добавьте таблицы **Покупатели**, **Продажа** и **Товар**. Выберите поля **Название Компании** из таблицы **Покупатели**, **Количество** из таблицы **Продажа** и **Цена** из таблицы **Товар**. В четвертом столбце на пустой ячейке в строке **Поле** установите курсор и вызовите Построитель выражений, щелкнув на кнопке **Построить** панели инструментов (на ней изображена волшебная палочка). На экране появится диалоговое окно Построителя выражений, показанное на рисунке 15.

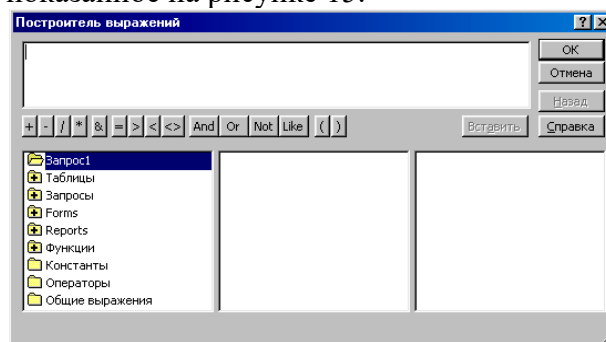


Рис. 15. Диалоговое окно Построителя выражений

Постройте выражение **Сумма: Format (Продажа!Количество *Товар !Цена)** так, как показано на рисунке 16.

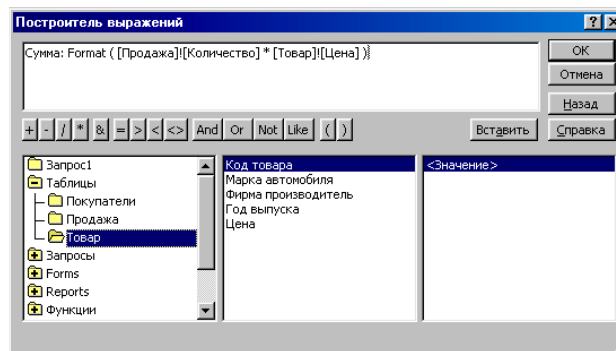


Рис. 16. Построение выражения для вычисляемого поля

После ввода выражения щелкните на кнопке **ОК** в диалоговом окне Построителя выражений. Access поместит созданное выражение в ячейку запроса. Сохраните запрос под именем **Сумма платежа**, просмотрите результаты запроса в режиме таблицы.

Создайте параметрический запрос с использованием вычисляемого поля. В таблице **Товар** для каждого из автомобилей указана его фиксированная цена (поле **Цена**). Предположим, что в рамках предпраздничных акций на все товары объявляется некоторая скидка, величина которой может изменяться. Чтобы вычислить новые цены на товары с учетом той или иной скидки, можно создать запрос с параметром, в роли которого и будет выступать величина скидки.

Для создания запроса запустите режим конструктора, добавьте таблицу **Товар** и выберите все поля данной таблицы. В шестом столбце на пустой ячейке в строке **Поле** установите курсор и вызовите Построитель выражений, щелкнув на кнопке **Построить** панели инструментов.

Задайте следующую формулу **Цена со скидкой**: $[Цена] * (1 - [Скидка \text{ в процентах}] / 100)$. Сохраните запрос под именем **Цена со скидкой**, запустите запрос.

Программа Access открывает диалоговое окно **Введите значение параметра**, куда необходимо ввести нужную для вычислений величину. Для удобства пользователя в рассматриваемом примере можно указывать величину скидки в процентах, а не в долях единицы (например, 5 или 20%, а не 0,05 или 0,2), этот нюанс учтен в расчетной формуле для поля **Цена со скидкой**.

Внимательно взглянув на результаты запроса, легко заметить курьезное несоответствие: цены на товары указаны в рублях, а цена со скидкой единиц измерения не имеет. Причиной является некорректный формат. Поскольку новое поле **Цена со скидкой** является вычисляемым, оно не наследует свойства исходной таблицы, и формат для него следует задавать непосредственно в запросе.

Чтобы изменить формат вычисляемого поля, выполните следующее. В окне Конструктора запроса щелкните правой кнопкой мыши на вычисляемом поле и выберите из открывшегося контекстного меню команду **Свойства** (или, поместив указатель в вычисляемое поле, щелкните на кнопке свойства панели инструментов Конструктор запросов). Откроется диалоговое окно **Свойства поля**, как показано на рисунке 17.

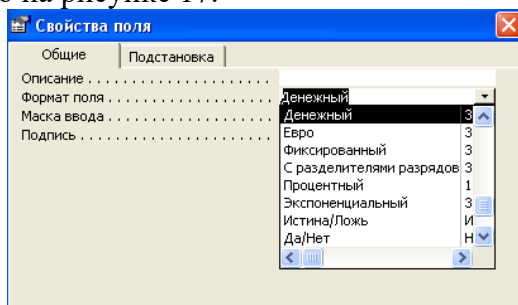


Рис. 17. Формат вычисляемых полей

Щелкните на поле опции **Формат**, а затем — на кнопке раскрывающегося списка и выберите подходящий формат для данных вычисляемого поля. Выполняя расчеты со значениями, которые содержатся в полях денежного типа, логично выбрать из списка формат **Денежный**.

Указав новый формат, закройте окно свойств и сохраните изменения запроса. Запустите запрос, все денежные единицы теперь указаны правильно.

Самостоятельная работа

1. Для чего применяются выражения в запросах?
2. Как создать вычисляемое поле в запросе: опишите работу Построителя выражений при создании запроса с вычисляемым полем.
3. Какова процедура изменения формата вычисляемого поля?

Лабораторная работа 23. Использование СУБД ACCESS для создания баз данных

Создайте автоформу для таблицы **Товар на сервисном ремонте**. Для этого нажмите на кнопку **Создать** окна базы данных.

На экране появится диалоговое окно **Новая форма**, где для категории автоформа доступны такие варианты, как в столбец, ленточная, табличная, сводная таблица и сводная диаграмма. Из списка, предложенного в окне Новая форма, выберите тип автоформы в столбец. Из раскрывающегося списка, расположенного в нижней части окна, выберите имя таблицы и щелкните затем на кнопке ОК. На экране появится выбранный вами тип формы, сохраните его под именем **Товар на ремонте**.

Создайте с помощью Мастера форму для таблицы **Товар**. Выберите все поля таблицы, внешний вид формы укажите в один столбец. Стиль формы выберите произвольно из предлагаемого списка, задайте имя формы **Товар** и просмотрите созданную форму. В созданной форме добавьте следующие данные: марка автомобиля – ВАЗ-2104, фирма

производитель – ВАЗ, год выпуска – 10.09.98, цена – 60 000 руб. Закройте форму, откройте таблицу **Товар** и убедитесь, что данные внесены в таблицу.

Конструкцию созданной Мастером формы можно изменить с помощью Конструктора. Для этого выберите форму **Товар** и на панели инструментов выберите режим **Конструктор** как на рисунке 18.

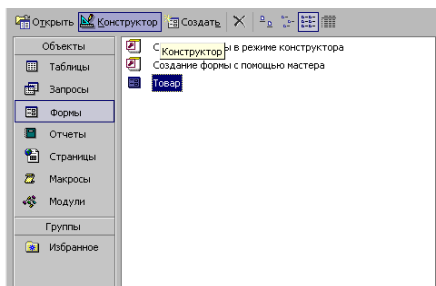


Рис. 18. Редактирование формы в режиме Конструктора

Откроется режим Конструктора и появится панель Инструментов конструктора, с помощью которой можно вставить графические объекты, рабочие кнопки и другие дополнительные элементы, усложняющие форму. В режиме Конструктора измените размеры формы, размеры полей, размер шрифта в полях должен быть 14, а также добавьте в форму кнопки: закрытие формы, первая запись, последняя запись, чтобы в итоге форма выглядела так, как это показано на рисунке 19.

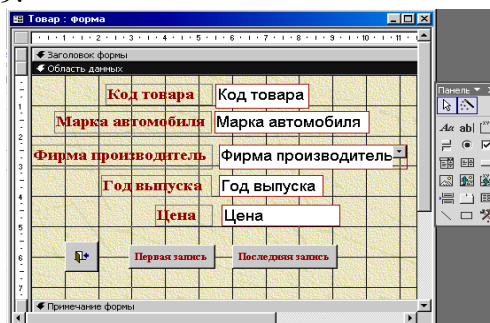


Рис. 19. Итоговая форма в окне Конструктора

Сохраните внесенные изменения формы, запустите форму и проверьте работоспособность кнопок.

Access позволяет создавать подчиненные формы. Подчиненная форма — это форма внутри формы, которая позволяет использовать данные из нескольких таблиц в одной форме. Между таблицами, для которых создается подчиненная форма, должна быть установлена связь данных.

Создайте для таблиц **Покупатели** и **Продажа** подчиненную форму. Для этого запустите создание формы в режиме Мастера, выберите таблицу **Покупатели**, добавьте все поля данной таблицы в поле Выбранные поля, потом в ячейке Таблицы и запросы выберите таблицу **Продажа** и аналогично добавьте все поля таблицы в поле Выбранные поля (таблица Покупатели считается главной, а таблица Продажа - подчиненной), после нажмите Далее. В следующем диалоговом окне выбирается тип представления данных. Предлагаются два варианта: 1) подчиненные формы – данные двух таблиц сразу будут доступны пользователю; 2) связанные формы – для получения доступа к данным подчиненной таблицы необходимо нажать на кнопку с названием подчиненной таблицы.

Выберите вариант подчиненные формы, в следующем диалоговом окне (вид подчиненной формы) укажите табличный. Стиль оформления выберите самостоятельно, имена форм оставьте без изменения, как предлагается автоматически. Нажмите Готово и просмотрите созданную форму. В списке форм отображается как сложная форма, так и отдельно форма подчиненной таблицы, каждую из них можно изменить с помощью режима конструктора.

Откройте форму **Покупатели** в режиме Конструктора и самостоятельно отредактируйте форму, чтобы в ней хорошо читались все надписи, а также добавьте кнопки: закрытие формы,

первая запись, последняя запись, предыдущая и следующая запись. Запустите форму и проверьте работоспособность кнопок.

Несмотря на то, что создание форм «с нуля» в режиме Конструктора — это самый сложный путь, но именно он позволяет реализовать весь потенциал, который заложен разработчиками Access в такие объекты базы данных, как формы.

Создайте с помощью Конструктора форму для таблицы **Продажа сотрудникам фирмы**. Оптимальным методом является использование кнопки **Создать**. На экране появится диалоговое окно **Новая форма**, в списке которого необходимо выбрать пункт **Конструктор**, а из раскрывающегося списка, расположенного в нижней части окна, выберите таблицу **Продажа сотрудникам фирмы**.

Панель элементов и список полей — это тот минимальный набор инструментов, с которого начинается создание формы в режиме Конструктора. Первое, что должно обязательно присутствовать в форме, — это поля таблицы, в которые заносятся данные.

Чтобы добавить в форму элемент управления, соответствующий полю, выполните следующее.

1. В окне списка полей щелкните на нужном поле и, удерживая кнопку мыши нажатой, перетащите поле в раздел **Область данных формы**. Указатель мыши при этом принимает вид микрополя. Когда кнопка отпущена, в форме появится элемент управления типа **Поле**, а слева от него — присоединенный заголовок. Такой заголовок является элементом управления **Надпись** и отображает либо имя поля таблицы, либо (если оно указано) значение свойства Подпись данного поля, дополненное справа двоеточием. Между полем таблицы и элементом управления Поле в форме теперь установлена связь.

2. Повторите шаг 1 требуемое число раз, чтобы перенести в форму все поля таблицы.

Увеличьте размер полей и шрифт до 14 pt. Добавьте в форму кнопки: **Закрытие формы**, **Открыть форму**, где укажите форму **Товар на ремонте**.

Итак, в форме размещены все необходимые элементы управления и надписи, т.е. в функциональном отношении форма готова к использованию, однако ее можно заметно преобразить. Инструментальные средства, доступные в режиме Конструктора, позволяют выделить элементы управления, разделы или группы элементов, добавить линии и прямоугольники, изменить цвет, применить такие эффекты, как затенение, объемное представление элементов или фоновые рисунки.

С помощью панели **Формат** задайте форме зеленый цвет фона, текст надписей сделайте красный, а для надписей примените вдавленное оформление.

Создание кнопочной формы при помощи Диспетчера кнопочных форм.

Для удобной работы с набором созданных элементов их следует объединить. Средством для этого является кнопочная форма.

Цель создания кнопочной формы — избавить конечного пользователя от необходимости открывать таблицы, запросы, отчеты. Одним нажатием кнопки он получает конечный результат — нужную ему информацию.

База данных, которая включает средства работы с данными и кнопочную форму для их вызова, может считаться законченным приложением.

Кнопочная форма, как это следует из ее названия, содержит кнопки. Она может быть украшена рисунком и надписями. При нажатии на кнопки происходят события, которые вызывают выполнение всех предусмотренных действий. Если это открытие другой кнопочной формы, то на последней должна быть кнопка возврата. Таким образом, создаются системы кнопочных форм, похожие на разветвленные меню.

Создайте кнопочную форму для данной базы данных, для этого выполните следующую последовательность действий.

1. Выберите пункт меню Сервис – Служебные программы – Диспетчер кнопочных форм.

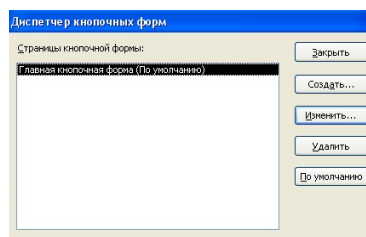


Рис. 20. Диспетчер кнопочных форм

2. Щелкните в окне Диспетчер кнопочных форм по кнопке **Изменить**.

3. В открывшемся окне Изменение страницы кнопочной формы щелкните по кнопке **Создать**. Появится диалоговая страница Изменение элемента кнопочной формы. Измените содержимое полей Текст, Команда, Форма в соответствии с рисунком.

Рис. 21. Изменение элемента кнопочной формы

4. Аналогично создайте еще два элемента кнопочной Формы: **Покупатели**, **Продажа**.

5. Добавьте кнопку закрытия базы данных. Для этого щелкните по кнопке **Создать**, в поле Текст выберите Выход, а в поле команда – **Выход** из приложения. Закройте диалоговые окна.

6. Откройте Кнопочную форму в режиме Конструктора, измените цвет и название вашей базы данных. Сохраните форму.

7. Проверьте работу формы.

Самостоятельная работа

1. Что такое формы и для чего они предназначены?
2. Какова процедура создания формы с помощью Мастера создания форм?
3. Какова процедура создания кнопочной формы?
4. Для чего предназначены подчиненные формы и как они создаются?

Лабораторная работа 24. Использование СУБД ACCESS для создания баз данных

С помощью Мастера отчетов создайте отчет по таблице **Товар**. Отчет должен содержать все поля таблицы. Установите уровень группировки отчета по полю Марка автомобиля. Записи в отчете должны быть отсортированы по полю Цена в порядке возрастания, макет запроса – ступенчатый, стиль отчета выберите самостоятельно. Задайте имя отчета **Товар** и просмотрите созданный отчет.

С помощью Мастера отчетов создайте отчет по запросу **Многопараметрический**, добавьте все поля запроса. Макет запроса выберите – блок, стиль отчета должен быть деловым. Сохраните отчет под именем **Многопараметрический** и проверьте работоспособность отчета.

Любой отчет, созданный с помощью Мастера, можно отредактировать в режиме Конструктора. Выберите отчет **Товар**, поместите его в режим Конструктора; надписи в отчете должны быть выровнены по центру, увеличьте шрифт, чтобы текст хорошо читался. Аналогично отредактируйте отчет **Многопараметрический**.

На панели инструментов есть кнопка **Создать**, предоставляющая дополнительные возможности. Рассмотрим ее функции на примере объекта отчет. С помощью данной кнопки можно создать автоотчет в столбец или ленточный, создать отчет с диаграммой, а также почтовые наклейки.

Создайте автоотчет в столбец для запроса **Автомобили ВАЗ**, ленточный автоотчет для таблицы **Покупатели**.

Возможность выбора полей не предусмотрена: в автоотчет помещаются все поля исходной таблицы или запроса. Хотя автоотчет работает только с одной таблицей, на

основании хорошо продуманного многотабличного запроса он позволяет быстро перенести на бумагу данные из нескольких таблиц.

Создайте отчет с почтовыми наклейками для всех зарегистрированных покупателей. Поскольку Мастер создания наклеек позволяет выбрать необходимые для отчета поля, можно не создавать специальный запрос, а работать непосредственно с таблицей Покупатели, в которой содержатся все нужные данные — имя покупателя, его код и адрес.

Для создания почтовой наклейки нажмите на кнопку **Создать** панели инструментов окна базы данных. На экране появится диалоговое окно **Новый отчет**, где представлен список всех доступных способов создания отчетов. Щелкните на пункте **Почтовые наклейки**, в нижней части окна выберите таблицу **Покупатели** и нажмите кнопку **ОК**. На экране появится первое окно Мастера создания наклеек, из раскрывающегося списка **Фильтр по изготовителю**, где перечислены самые известные производители бумаги для наклеек, выберите нужный пункт (например, Avery), а затем — подходящий размер наклейки (например, для кода M88315 это 34 mm x 64 mm, число наклеек по горизонтали — 3). Чтобы перейти к следующему окну Мастера, щелкните на кнопке **Далее**.

Новое окно Мастера позволяет выбрать шрифт и цвет для наклейки. Параметры, заданные в этом окне, применяются ко всем строкам будущей наклейки и ко всем наклейкам, которые будут распечатаны в этом отчете. Выберите подходящие значения для полей шрифт, размер, насыщенность и цвет текста, а также установите в случае необходимости флажки опций курсив и подчеркнутый. Слева от группы **Вид текста** расположен образец будущей наклейки, соответствующий выбранным настройкам шрифта. Убедившись, что внешний вид новой наклейки вас устраивает, щелкните на кнопке **Далее**.

На этом этапе необходимо указать, какие данные из таблицы-источника должны быть включены в наклейку. Слева в диалоговом окне Мастера представлен раскрывающийся список **Доступные поля**, справа — область **Прототип наклейки**. Количество строк в этой области зависит от размеров самой наклейки и выбранного шрифта. Включите в наклейку все поля из списка **Доступные поля**. Указав все необходимые поля, нажмите на кнопку **Далее**.

Следующее окно Мастера позволяет выбрать порядок сортировки будущих наклеек по одному или нескольким полям. Дважды щелкните на имени **Название компании** в списке **Доступные поля**, чтобы поместить его в список **Сортировка**. Выполнив нужные настройки, щелкните на кнопке **Далее**.

В заключительном окне Мастера введите в соответствующее текстовое поле нужный заголовок для отчета с наклейками **Наклейки Покупатели**, примите предложенную по умолчанию опцию **Просмотреть наклейки** в том виде, как они будут напечатаны, и щелкните на кнопке **Готово**.

Отчет может быть представлен в виде диаграммы. В диалогом окне кнопки **Создать** выберите Мастер диаграмм и укажите таблицу **Продажа**, сделайте доступными поля код покупателя и количество. Сохраните отчет под именем **Продажа**. В данном отчете в режиме Конструктора увеличьте размеры диаграммы, чтобы она стала нагляднее, как, например, показано на рисунке 20.

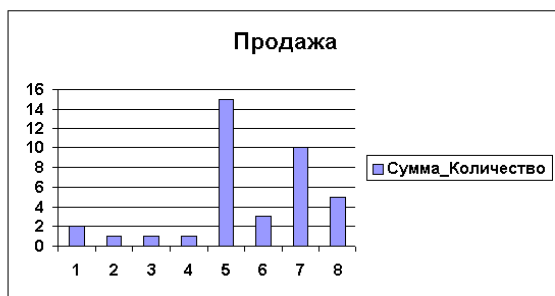


Рис. 22 Диаграмма отчета Продажа.

7 семестр

Лабораторная работа № 1. Создание макросов

Задание 1. Создайте макросы. Переключение в режим разметки страницы, переход на альбомную ориентацию, отключение линейки и установление масштаба отображения по ширине окна документа. Назначьте комбинацию клавиш.

Задание 2. Создайте макрос. Создайте кнопки: вставляющие самолетик, смайлик, заранее скопированный логотип физмата, рамочку из тортиков для каждой из страниц.

Лабораторная работа № 2. Создание макросов

Задание 1. Создать макрос в Microsoft Excel, который предназначен для копирования выделенной строки из одного листа (пусть это будет некий прайс-лист) в текущую строку (начиная с выделенной ячейки) другого листа книги Excel. После работы макроса на экране должен быть отображен лист, с которого осуществлялось копирование. Назначить вызов макроса по нажатию комбинации клавиш Ctrl + Й.

Задание 2. Макрос закрашивает желтым цветом 5 ячеек, расположенных ниже выделенной.

Задание 3. Создайте таблицу по образцу. Рассчитайте столбец Долг и итоговые значения в столбцах.

Создайте макрос. Макрос добавляет строку в нижнюю часть таблицы, перед итоговыми значениями, копирует формулу в новую ячейку столбца Долг, ставит порядковый номер строки. Проверьте правильность расчета итоговых значений в столбцах.

№	Фирма	№ Счета	Сумма в счете	Сумма оплаты	Долг
1	Ландыш	11	100	100	0
2	Василек	21	200	200	0
3	Ромашка	31	300	150	150
4	Ландыш	12	400	400	0
			1000	850	150

Задание 4. Создайте макрос, который по выделенной таблице строит график.

Задание 5. Управление макросами. Вкладка ленты Сервис в Microsoft Word и Microsoft Excel содержит кнопку Макросы. Сравните управление макросами.

Задание 6. Вынести на панель инструментов кнопки для реализации каждого макроса.

Лабораторная работа № 3. Макрорекордер в PowerPoint 2003

Задание 1. Макрос должен покрасить выделенную надпись в зеленый цвет. Вынести кнопку в собственную настраиваемую панель? Нарисовать на кнопке собственный рисунок. (При открытой настройке, правой кнопкой мыши по кнопке / изменить значок на кнопке.) В каком режиме работает макрос?

Задание 2. Создать макрос, который добавляет пустой слайд в презентацию, поместить его под управляющую кнопку. (Показ слайдов/ управляющие кнопки / Запуск макроса) В каком режиме работает макрос?

Задание 3. Создайте макрос создающий звезду красного цвета.

Задание 4. Создать управляющую кнопку, которая содержит макрос меняющий цвет всех слайдов на зеленый. А также вынести его на панель инструментов. (На слайде правой кнопкой мыши / Фон) В каком режиме работает макрос?

Задание 5. Создайте макрос, который на каждый слайд помещает дату и номер слайда.

Задание 6. Создайте макрос, который добавляет на слайд звук аплодисментов.

Задание 7. Создайте макрос, который дублирует слайд.

Задание 8. Создайте макрос, который делает выбранный слайд скрытым.

Задание 9. Создайте макрос, который добавляет на слайд управляющую кнопку далее, меняет ее цвет, прозрачность, добавляет на нее надпись и тень.

Задание 10. Создайте 5 собственных макросов, два из которых должны выполняться по щелчку на управляющую кнопку.

Лабораторная работа № 4. Создание кнопок на основе подсказок из макросов.

1. В рабочем окне документа Word создайте кнопку, которая будет менять при нажатии свой цвет и размер случайным образом.
2. Создайте кнопку в Excel «Стипендия», которая при попытке нажать будет менять цвет и местоположение.
3. Создать 2 кнопки в документе Word, которые по нажатию будут увеличивать документ, менять цвет фона на желтый и уменьшать документ, менять цвет документа на синий.
4. При загрузке документа, должно появляться окно с приветствием, а при закрытии с прощанием.
5. Создать макрос, который здоровается, при запуске документа и сразу его закрывает.
6. Добавить после приветствия запрос пароля, если пароль неверный закрыть документ.

Лабораторная работа №5. Создание и выполнение макросов

1. Создадим простой макрос, который изменяет шрифт, цвет заливки и направление текста в ячейке.
2. Разработаем в Excel таблицу расчета заработной платы (ФИО, начислено, налог, к выдаче) и создадим два макроса. Первый макрос будет отвечать за расчет значений в графах «Налог» и «К выдаче», а также итоговых сумм по полям «Начислено», «Налог», «К выдаче», а второй – за оформление таблицы.

Лабораторная работа №6. Редактирование макросов

В соответствии с вариантом (№ п/п) создать макрос, определить комбинацию клавиш для его вызова и назначить макрос графическому объекту. Проверить работу макроса. Просмотреть код макроса и отредактировать его, изменив значение одного из свойств, после чего снова запустить макрос на выполнение.

№ п/п	Описание макроса
1	Макрос, создающий копию рабочего листа «Лист1», помещаемую после рабочего листа «Лист2».
2	Макрос, устанавливающий в ячейке выравнивание по правому краю по горизонтали, по центру по вертикали, а направление текста – «снизу вверх».
3	Макрос, производящий автозаполнение ячеек столбца названиями дней двух недель, начиная с понедельника, и изменяющий цвет заливки этих ячеек на бирюзовый.
4	Макрос, объединяющий содержимое двух ячеек столбца, устанавливающий выравнивание по правому краю по горизонтали и изменяющий цвет символов.
5	Макрос, устанавливающий в ячейке числовой формат с тремя десятичными знаками, выравнивание вверх по центру.
6	Макрос, центрирующий по горизонтали содержимое ячейки и изменяющий размер и цвет символов в ячейке.
7	Макрос, устанавливающий цвет фона в семи соседних ячейках в соответствии с цветами радуги.
8	Макрос, преобразующий данные в ячейке в формат времени (минуты и секунды) и изменяющий размер символов на более крупный.
9	Макрос, устанавливающий в ячейке направление текста «сверху вниз», центрирующий содержимое ячейки по горизонтали и вертикали и изменяющий тип шрифта.
10	Макрос, добавляющий ниже выделенной ячейки строку, а правее – столбец и изменяющий цвет фона данной ячейки.
11	Макрос, производящий автозаполнение ячеек строки названиями месяцев года, начиная с сентября, и устанавливающий в этих ячейках начертание шрифта «полукурсив».
12	Макрос, устанавливающий в ячейке формат даты «14 мар 09», изменяющий цвет фона и цвет символов.
13	Макрос, устанавливающий цвет символов в семи соседних ячейках в соответствии с цветами радуги.
14	Макрос, объединяющий содержимое трех ячеек строки, устанавливающий выравнивание по горизонтали по левому краю и изменяющий размер символов на более крупный.
15	Макрос, преобразующий данные в ячейке в формат даты (число, месяц, год, часы, минуты) и центрирующий содержимое ячейки.

Лабораторная работа №7. Создание пользовательских функций MS EXCEL

1. Создадим пользовательскую функцию, вычисляющую по формуле Пифагора длину гипотенузы прямоугольного треугольника.
2. Создадим функцию пользователя, математически определенную как и построим ее график. $y = \sin(\pi x)e^{-2x}$

Лабораторная работа №8. Создание пользовательских функций MS EXCEL

В соответствии с вариантом (№п/п) создать пользовательскую функцию MS Excel и построить ее график.

№ п/п	Функция	№ п/п	Функция
1	$y = \sin(3\pi x) - 2\sin(5\pi x)$	2	$y = 2\cos(\pi x)\sin(3\pi x) + \sin(\pi x)$
3	$y = 3\sin(0,5\pi x)\sin(0,3\pi x) - 0,5\cos(\pi x)$	4	$y = 0,4\cos(\pi x)\sin(\pi x)\cos(3\pi x)$
5	$y = 2\cos(3\pi x)\sin(\pi x) + \frac{\sin(3\pi x)}{x}$	6	$y = 2\sin(\pi x) + \frac{\sin(3\pi x)}{3x}$
7	$y = 3\cos(\pi x)\sin(3\pi x)\sin(\pi x)$	8	$y = 5\sin(3\pi x) + 2\sin(5\pi x) - 3$
9	$y = \frac{\cos(\pi x)}{x} + x\sin(\pi x)$	10	$y = \frac{\cos(\sin(5\pi x))}{x} - \cos(\pi x)$
11	$y = \cos(3\pi x) + \sin(5\pi x)$	12	$y = \cos(\pi x)\sin(\pi x) - \cos(3\pi x)$
13	$y = \frac{\cos(\pi x) + \sin(\pi x)}{x}$	14	$y = \frac{4\cos(\pi x)\sin(5\pi x)}{x} - x$
15	$y = 2\sin(\pi x)\sin(3\pi x) - 0,5$		

Лабораторная работа № 9. Разработка пользовательских форм с помощью VBA В MS EXCEL

Используя VBA в MS Excel, разработаем пользовательскую форму «Расчет прибыли» для расположенной ниже таблицы.

Выручка от реализации (ВР)	Себестоимость (С)	Внереализационный доход (ВД)	Балансовая прибыль (БП)	Налог на прибыль (НП)	Сумма налога (СН)	Размер прибыли (РП)
900,00 р.	400,00 р.	150,00 р.	(ВР + ВД – С)	20,0 %	(БП · НП)	(БП – СН)

Лабораторная работа № 10. Разработка пользовательских форм с помощью VBA В MS EXCEL

Используя VBA в MS Excel, разработать пользовательскую форму, предназначенную для выполнения расчетов, соответствующих варианту задания (№ п/п). Форма должна содержать:

1. Заголовок формы.
2. Надписи и текстовые поля, соответствующие наименованиям столбцов таблицы.
3. Кнопку «Расчет», при нажатии на которую изменяется цвет ее фона, считывается исходная информация из соответствующих текстовых полей, производятся необходимые вычисления и вывод результатов в предназначенные для них текстовые поля.
4. Кнопку «Заполнить таблицу», при нажатии на которую изменяется цвет ее фона и заполняются соответствующие ячейки строки, расположенной ниже шапки таблицы на листе рабочей книги MS Excel (шапка таблицы включает все графы, а в ячейках строки, расположенной ниже, устанавливаются соответствующие форматы).
5. Кнопку «Очистить», при нажатии на которую удаляется информация, содержащаяся в текстовых полях формы и ячейках строки, расположенной ниже шапки таблицы.
6. Кнопку «Выход», предназначенную для закрытия формы.

Лабораторная работа № 11. Программирование разветвляющихся вычислительных процессов

Брокер получает процент от суммы сделки. Для сделок, сумма которых менее 150 тыс. р., вознаграждение брокера составит 3,5 %, от 150 до 500 тыс. р. – 4,5 %, а для сделок, сумма которых составляет больше 500 тыс. р. – 5 %. Разработать пользовательскую форму и составить программу для расчета вознаграждения брокера.

Лабораторная работа № 12. Свойство "Обработка событий" для элементов управления. Операторы If, Select Case.

Создайте новую базу данных. В ней простую форму в режиме конструктора с именем "Упражнение №1".

Задание №1. В форме создать кнопку, при нажатии которой в поле надписи высвечивается текущая дата.

Порядок выполнения. Внутри формы создайте надпись, присвойте ей имя посредством свойства "Имя". Значением имени пусть будет слово "Сегодня". Создайте кнопку с текстом, который приведен на рисунке. Для кнопки определите свойства "События" - "Нажатие кнопки" как процедуру обработки события. Программа на VBA будет по умолчанию иметь имя "Кнопка№_Click". В ее тексте запишите строку:

```
Me.Сегодня.Caption = Date()
```

Таким образом, функция даты (Date) присвоит значение надписи (Сегодня.Caption) Запустите программу, исправьте возможные ошибки.

Задание №2. В форме создать кнопку, при нажатии которой в поле надписи может быть введена произвольная дата.

Порядок выполнения. Аналогично предыдущему примеру вплоть до текста процедуры обработки события. Если имя надписи "Всегда", то текстом будет:

```
Dim d As String
```

```
d = InputBox("Введите дату")
```

```
Me.Всегда.Caption = d
```

Объявлена переменная d для ввода даты.

Окно ввода данных InputBox

Надпись приняла Caption, равное d.

Если выполнить отмену ввода даты, то переменная d не принимает никакого значения, что ведет к ошибке. Добавьте строку, которая прервет выполнение события (Exit Sub), если d - не дата (Not IsDate(d)):

```
If Not IsDate(d) Then Exit Sub
```

Задание №3. Надпись высвечивается цветом, принятым по умолчанию (черный). Изменить цвет надписи, если введенная дата - выходной день (воскресенье).

Порядок выполнения. Внесите изменения в текст процедуры обработки события:

После ввода даты определите номер дня в неделе, считая, что понедельник - 1-ый.

```
Dim n As Integer
```

```
n = WeekDay(d, vbMonday)
```

Выберите цвет надписи (ForeColor) равным 255 (красный), если номер = 7, и 0(черный) в остальных случаях.

```
If n = 7 Then
```

```
Me.Всегда.ForeColor = 255
```

```
Else
```

```
Me.Всегда.ForeColor = 0
```

```
End If
```

```
Me.Всегда.Caption = d
```

Дополнительное задание: Измените цвет и для субботы (шестой день недели).

Задание №4. Создать кнопку, при нажатии которой будет определен день недели для введенной даты.

Порядок выполнения. Аналогично второму заданию. Вместо вывода значения даты в надпись (имеет имя "Неделя"), нужно выводить текст названия дня недели в зависимости от его номера, который известен переменной n (например, номер понедельника = 1).

Для выбора названия по номеру используйте оператор Select Case:

```
Select Case n
```

```
Case 1
```

```
Me.Неделя.Caption = "Понедельник"
```

```
Case 2
```

Хочешь узнать
какое сегодня
число? кликни
здесь

12.11.2013

Хочешь ввести
дату? кликни
здесь

7.10.2013

Хочешь узнать
день
недели? кликни
здесь

Среда

Me.Неделя.Caption = "Вторник"
< и так далее все семь дней недели >
End Select

Лабораторная работа №13,14. Элементы управления "переключатель" и "флажок". Группа переключателей. Свойство "Обработка событий" для группы, для флажка. Операторы If, Select Case.

Объединение действий выполняется кнопками, которые могут быть трех типов: переключатель, флажок и выключатель. Их можно объединить в группу. Эти элементы управления имеют такие же названия.

Создайте новую форму с именем "Упражнение №2". Уберите из макета все элементы, связанные с обработкой данных (полосы прокрутки, области выделения, кнопки перехода.) Правой кнопкой мыши / Свойства

Задание №1. Создать в форме группу для вывода даты с двумя переключателями, (значок на панели элементов) выбор одного из которых позволит дату вывести как текущую, а другого - определить вводом.

Подпись группы - "Работа с датой".

Порядок выполнения.

В режиме конструктора в области данных новой формы создайте группу (имя группы по умолчанию будет "Группа0"), содержащую два переключателя. Надписи переключателей - "Текущая" и "Ввести". По умолчанию активна "Текущая". Значения переключателей оставьте по умолчанию 1,2. Тип элементов группы - переключатели. Настройте оформление. Надпись, именуемая группой, размещена на рамке группы. Измените ее содержание на "Работа с датой".

Для того чтобы переключатель изменял значение даты, необходимо добавить внутрь группы надпись, в которой и будет отображаться значение даты. Имя надписи - "Дата".

Программа на VBA должна обрабатывать событие "После обновления" из свойств группы ("Группа0"). Процедура будет называться Группа0_AfterUpdate() и будет обновлять значение надписи каждый раз при изменении значения переключателя (AfterUpdate).

Смысл и содержание программы - как в пп. 1 и 2 лабораторного задания №1. Отличия два:

1. При выборе переключателя "Текущая" значение Me.Группа0.Value = 1, при выборе "Ввести" =2. Проверку выполните конструкцией If Then. В первом случае выводите в надпись "Дата" (Caption) значение текущей даты (Date), во втором - введенное через InputBox.

2. Оба переключателя изменяют значение одной и той же надписи ("Дата").

Задание №2. Добавить флажок для вывода дня недели независимо от способа выбора даты (его можно выводить или нет).

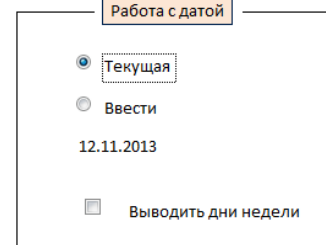
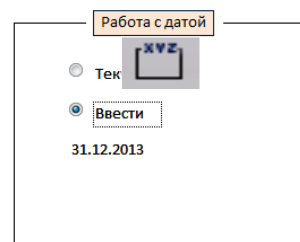
Текущая дата - по умолчанию. Логически флажок должен быть внутри группы, но группа может работать только с элементами одного типа, поэтому флажок с надписью "Выводить день недели" создайте вне группы, а надпись для вывода дня недели - внутри. Границу группы сделайте невидимой, а все элементы управления обведите вручную, как

в примере, с помощью линий (значок  на панели инструментов).

Надпись для дня недели внутри группы пусть имеет имя "День".

Значение даты для отображения дня недели следует взять из надписи "Дата" (Me.Дата.Caption). Для правильного отображения она не должна быть пустой, значит, по умолчанию там должна быть текущая дата. Для этого в свойствах всей формы событие "Открытие", соответствующее открытию формы (Form_Open), должно показать текущую дату в надписи "Дата", как в п.1 лабораторного задания №1.

Программа на VBA должна обрабатывать событие "После обновления" из свойств флажка ("Флажок№").



Процедура будет называться `Флажок№_AfterUpdate()`. Чтобы значение даты получить правильно, начните процедуру с получения даты:

```
Dim d As Date
```

```
d = Me.Дата.Caption
```

Таким образом, будет получена дата независимо от выбора способа ее задания в группе. Далее необходимо сделать надпись "День" видимой, если флажок включен (`Me.День.Visible = True`) и невидимой (`Me.День.Visible = False`), в обратном случае. Для определения дня недели используйте инструкцию `Select Case`, как в п.4 лабораторного задания №1.

Дополнительное задание. Задайте выделение цветом названий дней недели для выходных дней (суббота, воскресенье), как в п.4 лабораторного задания №1.

Лабораторная работа №15. Свойство "Обработка событий" для элементов управления и форм. Операторы VBA.

Откройте рабочую базу данных. Проект будет посвящен решению задачи обработки информации для некоторой фирмы, которая занимается обучением, набирая слушателей на разнообразные курсы. Первая часть проекта - "Сотрудники". Сотрудники не являются штатными, это информация о контингенте преподавателей, которые могут выполнять разовые работы. Преподаватель может временно не проводить занятий или проводить сразу на нескольких курсах. Контракт заключается на время выполнения работ и прекращается с их завершением.

Создайте таблицу с именем "Сотрудники". В качестве имен полей используйте простые слова без знаков подчеркивания и прочих знаков препинания.

№	Фамилия И.О.	Основное место работы	Должность	Телефон	Тематика занятий	Год рождения	Срок начала контракта	Фотография
1	Иванов Иван Петрович	ЮУрГУ, кафедра информатики	Профессор	39-95-22	Компьютерный дизайн. Работа в Internet.	1951		
2	Егорова Алена Вадимовна	Фабрика художественных изделий.	Художник	12-21-22	Плетение на коклюшках	1978	3.02.2000	

Для ввода и редактирования данных о сотрудниках необходимы две формы:

1. **Ленточная**, в которой отображены только поля таблицы:

2. "Номер", "Фамилия", "Тематика", "Год". Используйте мастер форм. Дайте форме имя "Сотрудники". Из макета формы уберите все элементы работы с записями, кроме полос прокрутки.

3. **Формуляр**. Формуляр используется для отображения и редактирования полной информации о сотруднике, в нем отражены все поля таблицы. Для перехода в режим работы с данными формуляра используйте в ленточной форме кнопку с надписью "Просмотр и редактирование текущей записи".

После редактирования записи или добавления новой кнопка "Сохранить" в формуляре занесет данные из полей формуляра в форму.

Порядок выполнения.

Формуляр создайте вручную в режиме конструктора. Его данные не должны быть привязаны к таблице, поэтому создайте 9 надписей в колонке слева и 9 свободных полей с именами, соответственно названиям полей таблицы. Они отражены в нижеприведенном тексте. В левой части присваиваний - Имена полей формы, в правой - значения.

```
Private Sub Кнопка22_Click()  
    DoCmd.OpenForm "Формуляр"  
    Form_Формуляр.Фамилия = Ме.Фамилия  
    Form_Формуляр.Телефон = Ме.Телефон  
    Form_Формуляр.Местоработы = Ме.Местоработы  
    Form_Формуляр.Должность = Ме.Должность  
    Form_Формуляр.Началоконтракта = Ме.Началоконтракта  
    Form_Формуляр.Год = Ме.Год  
    Form_Формуляр.Фотография = Ме.Фотография  
    Form_Формуляр.Номер = Ме.Номер  
    Form_Формуляр.Тематика = Ме.Тематика  
End Sub
```

Кнопка "Сохранить" в формуляре должна занести все введенные данные в таблицу, и закрыть формуляр.

```
Private Sub Кнопка20_Click()  
    Form_Сотрудники.Фамилия = Ме.Фамилия  
    Form_Сотрудники.Телефон = Ме.Телефон  
    Form_Сотрудники.Местоработы = Ме.Местоработы  
    Form_Сотрудники.Должность = Ме.Должность  
    Form_Сотрудники.Началоконтракта = Ме.Началоконтракта  
    Form_Сотрудники.Год = Ме.Год  
    Form_Сотрудники.Фотография = Ме.Фотография  
    Form_Сотрудники.Номер = Ме.Номер  
    Form_Сотрудники.Тематика = Ме.Тематика  
    Form_Сотрудники.Refresh  
    DoCmd.Close  
End Sub
```

8 семестр

Лабораторная работа №1-2. Комбинаторные и рекурсивные алгоритмы

Задания для решения

1. Разработать программу генерации всех последовательностей длины k из чисел $1, 2, \dots, N$. Первой последовательностью является $1, 1, \dots, 1$, последней – $N, N, \dots, N$.

2. Разработать программу генерации всех последовательностей длины k , у которых i -й элемент не превосходит значения i . Первой последовательностью является $1, 1, \dots, 1$, последней – $1, 2, \dots, k$.

3. Перечислить все разбиения натурального числа N на натуральные слагаемые (разбиения, отличающиеся лишь порядком слагаемых, считаются за одно) в следующих порядках (пример, при $N=4$):

- 4, 3+1, 2+2, 2+1+1, 1+1+1+1;
- 4, 2+2, 1+3, 1+1+2, 1+1+1+1;
- 1+1+1+1, 1+1+2, 1+3, 2+2, 4.

4. Разработать программу генерации всех последовательностей длины $2*N$, составленных из N единиц и N минус единиц, у которых сумма любого начального отрезка неотрицательна, т. е. количество минус единиц в нем не превосходит количества единиц.

5. На плоскости заданы N точек с целочисленными координатами. Требуется выделить те точки этого множества, которые образуют выпуклую оболочку (вырожденные случаи – все точки на одной прямой и т. д. не рассматриваются).

Указание. В данном случае можно воспользоваться следующим фактом. Точка (x, y) принадлежит выпуклой оболочке, если она не лежит внутри любого треугольника, образованного остальными точками исходного множества.

Лабораторная работа № 3-6. Алгоритмы поиска

Задания для решения

1. Задача о парламенте*. На острове Новой Демократии каждый из жителей организовал партию, которую сам и возглавил. Любой из жителей острова может состоять не только в своей партии, но и в других партиях. К всеобщему удивлению, даже в самой малочисленной партии оказалось не менее двух человек. К сожалению, финансовые трудности не позволили создать парламент, куда вошли бы, как предполагалось по Конституции острова, президенты всех партий. Посоветовавшись, островитяне решили, что будет достаточно, если в парламенте будет хотя бы один член каждой партии.

Помогите островитянам организовать такой, как можно более малочисленный парламент, в котором будут представлены все партии.

Исходные данные: каждая партия и ее президент имеют один и тот же порядковый номер от 1 до N ($4 \leq N < 150$). Вам даны списки всех N партий острова Новой Демократии. Выведите предлагаемый вами парламент в виде списка номеров ее членов.

Например, для четырех партий: Список членов парламента 2 состоит из одного жителя с номером 2.

2. Какое наименьшее число ферзей можно расставить на доске так, чтобы они держали под боем все ее свободные поля?

Модификация задачи. Найти расстановку ферзей, которая одновременно решает задачу для досок 9×9 , 10×10 и 11×11 .

Указание. Задачу можно решать как обычным перебором, так и, представив доску как граф, путем поиска минимального доминирующего множества вершин.

3. Расставить на доске $N \times N$ ($N < 12$) N ферзей так, чтобы наибольшее число ее полей оказалось вне боя ферзей.

4. Расставить на доске как можно больше ферзей так, чтобы при снятии любого из них появлялось ровно одно неатакованное поле.

Лабораторная работа №7-10. Алгоритмы сортировки

Задания для решения

1. Отсортировать элементы данного одномерного числового массива методом «пузырька».

2. Отсортировать элементы данного одномерного числового массива методом простого выбора.

3. Отсортировать элементы данного одномерного числового массива методом Шелла.

4. Отсортировать элементы данного одномерного числового массива методом «быстрой» сортировки.

5. Отсортировать элементы данного одномерного числового массива методом пирамидальной сортировки.

6. Отсортировать элементы данного одномерного числового массива методом подсчета.

7. Отсортировать элементы данного одномерного числового массива методом сортировки с убывающим шагом.

8. Отсортировать элементы данного одномерного числового массива методом быстрой сортировки с выбором элемента x случайным образом.

9. Отсортировать элементы данного одномерного числового массива методом шейкерной сортировки.

Лабораторная работа №11-12. Алгоритмы на графах

Задания для решения

1. Дан ориентированный граф с N вершинами ($N \leq 50$). Вершины и дуги окрашены в цвета с номерами от 1 до M ($M \leq 6$). Указаны две вершины, в которых находятся фишки игрока и конечная вершина. Правило перемещения фишек: игрок может передвигать фишку по дуге, если ее цвет совпадает с цветом вершины, в которой находится другая фишка; ходы можно делать только в направлении дуг графа; поочередность ходов необязательна. Игра заканчивается, если одна из фишек достигает конечной вершины. Написать программу поиска кратчайшего пути до конечной вершины, если он существует.

2. Некоторые школы связаны компьютерной сетью. Между школами заключены соглашения: каждая школа имеет список школ-получателей, которым она рассылает программное обеспечение всякий раз, получив новое бесплатное программное обеспечение (извне сети или из другой школы). При этом, если школа B есть в списке получателей школы A , то школа A может не быть в списке получателей школы B .

Требуется написать программу, определяющую минимальное количество школ, которым надо передать по экземпляру нового программного обеспечения, чтобы распространить его по всем школам сети в соответствии с соглашениями.

Кроме того, надо обеспечить возможность рассылки нового программного обеспечения из любой школы по всем остальным школам. Для этого можно расширять списки получателей некоторых школ, добавляя в них новые школы. Требуется найти минимальное суммарное количество расширений списков, при которых программное обеспечение из любой школы достигло бы всех остальных школ. Одно расширение означает добавление одной новой школы-получателя в список получателей одной из школ.

3. Задан неориентированный граф. При прохождении по некоторым ребрам отдельные (определенные заранее) ребра могут исчезать или появляться. Найти кратчайший путь из вершины с номером q в вершину с номером w .

4. Заданы два числа N и M ($20 \leq M \leq N \leq 150$), где N – количество точек на плоскости. Требуется построить дерево из M точек так, чтобы оно было оптимальным. Дерево называется оптимальным, если сумма всех его ребер минимальна. Все ребра – это расстояния между вершинами, заданными координатами точек на плоскости.

5. Даны два числа N и M . Построить граф из N вершин и M ребер. Каждой вершине ставится в соответствие число ребер, входящих в нее. Граф должен быть таким, чтобы сумма квадратов этих чисел была минимальна.

6. Задан ориентированный граф с N вершинами, каждому ребру которого приписан неотрицательный вес. Требуется найти простой цикл, для которого среднее геометрическое весов его ребер было бы минимально.

7. Ребенок нарисовал кружки и некоторые из них соединил отрезками. Кружки он пометил целыми числами от 1 до N ($1 \leq N \leq 30$), а на каждом отрезке поставил стрелочку. Затем он приписал каждому кружочку вес в виде некоторого целого числа и определил начальный и конечный кружочки. Из первого он должен выйти, а во второй попасть.

Ребенок решил для себя следующее:

- набрать максимально возможное суммарное количество очков;
- по каждому отрезку пройти ровно один раз;
- если в кружок он попадает при движении по направлению стрелки, то к суммарному количеству очков вес этого кружка прибавляется;
- если в кружок он попадает при движении против направления стрелки, то из суммарного количества очков вес того кружка вычитается.

Написать программу, которая бы помогла ребенку построить путь, удовлетворяющий всем этим требованиям.

Лабораторная работа №13-16. Задачи на клеточной карте

Задания для решения

1. На клеточной карте заданы острова (-1) и водная поверхность. Пользователь вводит координаты исходной и конечной точки маршрута (движение возможно только по воде). Требуется проложить путь между заданными точками. Движение можно осуществлять по горизонтали либо по вертикали. Определить длину маршрута и выделить путь цветом.

2. Клетки доски 8×8 раскрашены в два цвета: белый и черный. Необходимо пройти из левого нижнего угла в правый верхний так, чтобы цвета клеток перемежались. За один ход разрешается перемещаться на одну клетку по вертикали или горизонтали.

Программа должна (если путь существует):

- находить хотя бы один путь;
- находить путь минимальной длины.

Указание. Для решения задачи применим метод «волны» с учетом условия задачи — чередования клеток разного цвета.

3. Возьмем клетчатую доску $M \times N$ ($1 \leq M, N \leq 4$). Воткнем в каждую клеточку штырек. В нашем распоряжении есть K ($0 < K < 20$) абсолютно одинаковых колечек, каждое из которых можно нанизывать на штырек, причем на один штырек разрешается надеть несколько колечек. Подсчитать, сколькими способами можно распределить все эти колечки по штырькам. Два распределения считаются разными, если на каком-то из штырьков находится разное количество колечек, и одинаковыми в противном случае.

Входные данные: M, N, K .

Пример: 2, 2, 2.

Результат: 10.

4. Дано прямоугольное клетчатое поле $M \times N$ клеток. Каждая клетка поля покрашена в один из шести цветов, причем верхняя левая и нижняя правая имеют различный цвет. В результате поле разбивается на некоторое количество одноцветных областей: две клетки одного цвета, имеющие общую сторону, принадлежат одной области.

Правила игры. За первым игроком закреплена область, включающая левую верхнюю клетку, за вторым – правую нижнюю. Игроки ходят по очереди. Делая ход, игрок перекрашивает свою область по своему выбору:

- в любой из шести цветов;
- в любой из шести цветов, за исключением цвета своей области и цвета области противника.

В результате хода к области игрока присоединяются все прилегающие к ней области выбранного цвета, если такие имеются. Если после очередного хода окажется, что области игроков соприкасаются, то игра заканчивается.

Написать программу, которая определяет минимально возможное число ходов, по прошествии которых игра может завершиться.

Входные данные. Цвета пронумерованы цифрами от 1 до 6. Первая строка входного файла (*input.txt*) содержит M и N – размеры поля ($1 \leq M, N \leq 50$). Далее следует описание раскраски поля – M строк по N цифр (от 1 до 6) в каждой без пробелов. Первая цифра файла соответствует цвету левой верхней клетки игрового поля. Количество одноцветных областей не превосходит 50.

Выходные данные. В выходной файл (*output.txt*) необходимо вывести искомое количество ходов для каждого из пунктов.

Лабораторная работа №17-20. Задачи на шахматной доске

Задания для решения

1. В некоторых клетках квадратной доски 4×4 находятся четырёхгранные игральные кости. Кость снимается с поля, если количество соседних по стороне непустых клеток совпадает с числом, выпавшим на кости. Все кости, для которых выполняется это свойство, снимаются с поля одновременно. Если после снятия костей появляются новые кости, которые

можно снять, то они снимаются по тем же правилам. Нужно переставить одну кость так, чтобы снялось максимальное количество костей.

На вход программе даётся четыре строки по четыре числа в каждом, разделённые одним пробелом. Все числа – целые неотрицательные, не превышающие 4. Ноль означает отсутствие кости, любое положительное – наличие кости, на которой выпало указанное число. Гарантируется, что на поле есть, по крайней мере, одна кость и одна свободная клетка.

Пример ввода 1:

```
0 0 1 0
0 0 0 0
0 0 1 0
0 0 0 0
```

Пример ввода 2:

```
0 0 0 1
0 2 1 0
0 1 2 1
0 0 0 0
```

В виде ответа нужно вывести одно целое число – количество костей, которое можно снять с доски.

Пример вывода 1:

2

Пример вывода 2:

6

2. Задача о коне Аттилы («Трава не растет там, где ступил мой конь!»). На шахматной доске стоят белый конь и черный король. Некоторые поля доски считаются «горящими». Конь должен дойти до неприятельского короля, повергнуть его и вернуться на исходное место. При этом ему запрещено становиться как на горящие поля, так и на поля, которые уже пройдены.

3. Магараджа – это фигура, которая объединяет в себе ходы коня и ферзя. Для доски 10×10 найти способ расстановки 10 мирных (не бьющих друг друга) магараджей.

4. Имеется механизм, состоящий из N расположенных в одной плоскости и свободно надетых на зафиксированные оси пронумерованных шестеренок, зубья которых соприкасаются друг с другом. Информация о механизме ограничивается тем, что для каждой шестеренки перечислены все те, с которыми она находится в непосредственном зацеплении. Напишите программу, определяющую, есть ли такая шестеренка, поворачивая которую мы приведем в движение весь механизм.

5. Дана доска 8×8 . На ней размещены один король и N коней ($0 < i < 63$). Фигуры перемещаются по шахматным правилам. В процессе перемещений в одной клетке разрешается размещать несколько фигур. Цель перемещений – собрать все фигуры в одной клетке, причем за минимальное общее количество перемещений. Дополнительное условие – если король и конь находятся в одной клетке, то их можно переместить вместе по правилу хода коня, причем это перемещение засчитывается за один ход.

Входные данные. Файл *input.txt* содержит одну строку символов без пробелов, описывающую начальное расположение фигур на доске. Строка содержит последовательность клеток доски, первая из которых – клетка короля, остальные – клетки коней. Каждая клетка описывается парой буква-цифра.

Выходные данные. Файл *output.txt* должен содержать одно число – минимальное количество перемещений, за которое все фигуры можно собрать в одной клетке доски.

9 семестр

Лабораторная работа 1-2. Основы разработки приложений в среде MIT App inventor

Разработайте простейшее приложение имитирующее игровой кубик, который будет выкидывать грань с нужным количеством кружков, если мобильное устройство тряхнуть.

Протестируйте и отладьте приложения на мобильном устройстве.

Лабораторная работа 3. Простейшие элементы управления в мобильных приложениях

Разработайте приложения, использующие простейшие элементы управления в мобильных устройствах:

1. Приложение “Загадка”.
2. Приложение “SoundBoard”.
3. Приложение “Отгадай-ка”.
4. Приложение “Виртуальный кот”.

Лабораторная работа 4. Многооконные приложения

Разработайте многооконное приложение “Сказочные превращения”.

Лабораторная работа 5. Обмен данными между экранами

Разработайте приложения, использующие обмен данными между экранами:

1. Приложение “Сказочные перемещения”.
2. Приложение “Хамелеон”.

Лабораторная работа 6. Списки

Разработайте приложения, использующее списки:

1. Создание собственного цвета.
2. Приложение “Фонарик”.
3. Приложение “Записная книжка”.
4. Приложение “Слайд-шоу”.

Лабораторная работа 7. Разработка графических приложений

Разработайте графические приложения:

1. Приложение “Рисование”.
2. Приложение “Пишем на холсте”.
3. Приложение “Конфетти”.

Лабораторная работа 8. Разработка приложений с элементами анимации

Разработайте приложения с элементами анимации:

1. Приложение “Игра в мяч”.
2. Приложение “Управляем движением объекта”.

Лабораторная работа 9. Медийные возможности MIT App inventor

Разработайте приложения, использующие медийные возможности MIT App inventor:

1. Приложение “Распознавание речи”.
2. Приложение “Испорченный телефон”.
3. Приложение “Переводчик”.
4. Приложение “Видеоплеер”.
5. Приложение “Мр3 плеер”.
6. Приложение “Фотокамера”.

Лабораторная работа 10. Разработка публицирующего приложения

Разработайте публицирующее приложение “Sharing”.

Лабораторная работа 11. Разработка ГЕО-приложений

Разработайте ГЕО-приложения:

1. Приложение “Где я?”.
2. Приложение “Компас”.

Лабораторная работа 12. Математические приложения

Разработайте математические приложения:

1. Приложение “Тренажер”.
2. Приложение “Конвертер систем счисления”.

Лабораторная работа 13. Разработка игрового приложения

Разработайте игровое приложение для определенной возрастной группы:

Первый период детства — от 3 до 7 лет

Второй период детства — от 8 до 12 лет

Подростковый возраст — от 13 до 16 лет

Юношеский возраст — от 17 до 21 года

Средний возраст

первый период — от 22 до 35 года

второй период — от 36 до 60 года

Пожилые люди — от 61 до 75 года

Лабораторная работа 14. Разработка справочного приложения по математике

Разработайте справочное приложение для подготовки к аттестации (ВПР, ОГЭ, ЕГЭ, ГВЭ) по математике.

Лабораторная работа 15. Разработка справочного приложения по информатике

Разработайте справочное приложение для подготовки к аттестации (ВПР, ОГЭ, ЕГЭ, ГВЭ) по информатике

Лабораторная работа 16. Обучающее приложение

Разработайте обучающее приложение по одному из разделов курса информатики в школе:

1. Информация и информационные процессы
2. Представление информации
3. Архитектура компьютера
4. Формализация и моделирование
5. Алгоритмизация и программирование
6. Информационные технологии
7. Социальная информатика

Самостоятельная работа

Текущая самостоятельная работа направлена на углубление и закрепление знаний и развитие практических умений. Она заключается в работе с теоретическим материалом, поиске и сборе литературы и источников информации по заданным разделам курса, подготовке к лабораторным работам, выполнение домашних заданий.

Задания для самостоятельного решения

Тема. *Решение прикладных математических задач*

Задания для решения

Вариант 1

Найти площадь криволинейной трапеции, ограниченной линиями $y=e^x$, $x=0$, $x=1$ (методом трапеций) при $n = 5, 10, 100, 1000$.

Найти сумму ряда с точностью 0,00001 и определить количество слагаемых этой суммы:

$$1/(1+2) + 1/(1+2+3) + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Вариант 2.

1. Найти значение выражения по схеме Горнера:

$$(p(2.7) + p(4))/2, \quad \text{где } p(x)=5x^5 + 9x^4 - 2.6x^3 + x^2 + 11.4x + 0.9;$$

2. Вычислить значение суммы с точностью $E=0.0001$

$$S = 1/2! + 1/3! + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Вариант 3.

1. Вычислить площадь криволинейной трапеции, ограниченной линиями: $y=(x-4)^2$, $x=1$, $x=4$. Выполнить вычисления для $n=100,500,1000$.

2. Вычислить значение суммы с точностью $E=0.0001$

$$S = 1/(1+2) - 1/(1+2+3) + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Вариант 4.

1. Найти значение выражения по схеме Горнера:

$$p(1.7) + p(4.8), \quad \text{где } p(x)=7x^5+9x^4-2.6x^3+3x^2+11.4x-0.9;$$

2. Вычислить значение суммы с точностью $E=0.00001$

$$S = 1/2! - 1/4! + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Вариант 5.

1. Вычислить площадь криволинейной трапеции, ограниченной линиями: $y=\ln(x-1)$, $x=2$, $x=4$. Выполнить вычисления для $n=100,200,1000$.

2. Вычислить значение суммы с точностью $E=0.000001$

$$S = 1/(1+2!) + 1/(2+3!) + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Вариант 6.

1. Найти значение выражения по схеме Горнера:

$$2p(7) + p(2)/2, \quad \text{где } p(x)=15x^5+9x^4-2x^3+x^2+11.4x-9;$$

2. Вычислить значение суммы с точностью $E=0.001$

$$S = 1/(1+3) - 1/(1+3+5) + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Вариант 7.

1. Вычислить площадь криволинейной трапеции, ограниченной линиями: $y=|2\cos x + 1|$, $x=0$, $x=\pi$. Выполнить вычисления для $n=100,500,1000$.

2. Вычислить значение суммы с точностью $E=0.00001$

$$S = 1/3! - 1/5! + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Вариант 8.

1. Найти значение выражения по схеме Горнера:

$$p(1.7) + p(8)/2, \quad \text{где } p(x)=19x^4-0.6x^3+5x^2+11.4x-10.9;$$

2. Вычислить значение суммы с точностью $E=0.00001$

$$S = 1/(2+3!) + 1/(3+5!) + \dots$$

Вариант 9.

1. Вычислить площадь криволинейной трапеции, ограниченной линиями: $y=|3\sin x-1|$, $x=\pi/2$, $x=2\pi$. Выполнить вычисления для $n=100,200,500$.

2. Найти сумму ряда с точностью 0,00001 и определить количество слагаемых этой суммы:

$$1/(2+4) + 1/(2+4+6) + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Вариант 10.

1. Найти значение выражения по схеме Горнера:

$$2p(0.7) + 3p(2), \quad \text{где } p(x)=5x^5+x^4-0.6x^3+2x^2+x-2.9;$$

2. Вычислить значение суммы с точностью $E=0.0001$

$$S = 1/2! - 1/4! + \dots$$

Для нахождения знаменателя напишите функцию пользователя.

Тема. Структуры данных

Задания для решения

Вариант 1.

1. Сформировать стек из 7 чисел. Найти сумму 5-го и 6-го элементов и поместить ее в стек.
2. Сформировать стек из 10 чисел. Найти количество положительных чисел стека.
3. Сформировать очередь из 7 чисел. Удвоить 4-й элемент и поместить его в очередь.
4. Сформировать очередь из 10 чисел. Найти среднее арифметическое четных чисел очереди.

Вариант 2.

1. Сформировать стек из 5 чисел. Найти произведение 4-го и 3-го элементов и поместить ее в стек.
2. Сформировать стек из 10 чисел. Найти количество четных чисел стека.
3. Сформировать очередь из 5 чисел. Найти произведение 1-го и 3-го элементов и поместить ее в очередь.
4. Сформировать очередь из 8 чисел. Найти сумму нечетных чисел очереди.

Вариант 3.

1. Сформировать стек из 6 чисел. Найти сумму 3-го и 5-го элементов.
2. Сформировать стек из 10 чисел. Найти количество отрицательных чисел стека и поместить в стек.
3. Сформировать очередь из 10 чисел. Удвоить сумму 4-го и 6-го элементов и поместить ее в очередь.
4. Сформировать очередь из 6 чисел. Найти среднее арифметическое чисел, кратных 4-м.

Вариант 4.

1. Сформировать стек из 5 чисел. Найти произведение 2-го и 3-го элементов и поместить ее в стек.
2. Сформировать стек из 8 чисел. Найти количество нечетных чисел стека и поместить в стек.
3. Сформировать очередь из 5 чисел. Найти произведение 1-го и 3-го элементов и поместить ее в очередь.
4. Сформировать очередь из 10 чисел. Найти количество четных чисел очереди.

Вариант 5.

1. Сформировать стек из 5 чисел. Удвоить сумму 3-го и 4-го элементов и поместить ее в стек.
2. Сформировать стек из 9 чисел. Найти среднее арифметическое положительных чисел стека.
3. Сформировать очередь из 6 чисел. Найти сумму 2-го и 4-го элементов.
4. Сформировать очередь из 10 чисел. Найти количество отрицательных чисел и поместить в очередь.

Вариант 6.

1. Сформировать стек из 7 чисел. Удвоить 4-й элемент и поместить его в стек.
2. Сформировать стек из 10 чисел. Найти среднее арифметическое четных чисел стека.

3. Сформировать очередь из 5 чисел. Найти произведение 2-го и 3-го элементов и поместить ее в очередь.
4. Сформировать очередь из 8 чисел. Найти количество нечетных чисел и поместить в очередь.

Вариант 7.

1. Сформировать стек из 6 чисел. Утроить сумму 3-го и 5-го элементов.
2. Сформировать стек из 10 чисел. Найти сумму отрицательных чисел стека и поместить в стек.
3. Сформировать очередь из 5 чисел. Удвоить сумму 3-го и 4-го элементов и поместить ее в очередь.
4. Сформировать очередь из 9 чисел. Найти среднее арифметическое положительных чисел очереди.

Вариант 8.

1. Сформировать стек из 6 чисел. Найти произведение 4-го и 5-го элементов и поместить ее в стек.
2. Сформировать стек из 8 чисел. Найти сумму нечетных чисел стека и поместить в стек.
3. Сформировать очередь из 5 чисел. Найти произведение 2-го и 3-го элементов и поместить ее в очередь.
4. Сформировать очередь из 10 чисел. Найти количество четных чисел очереди.

Вариант 9.

1. Сформировать стек из 7 чисел. Уменьшить на единицу 4-й элементов и поместить его в стек.
2. Сформировать стек из 9 чисел. Найти среднее арифметическое чисел, кратных трем.
3. Сформировать очередь из 6 чисел. Утроить сумму 3-го и 5-го элементов.
4. Сформировать очередь из 10 чисел. Найти сумму отрицательных чисел и поместить в очередь.

Вариант 10.

1. Сформировать стек из 10 чисел. Удвоить сумму 4-го и 6-го элементов и поместить ее в стек.
2. Сформировать стек из 6 чисел. Найти среднее арифметическое чисел, кратных 4-м.
3. Сформировать очередь из 6 чисел. Найти произведение 3-го и 5-го элементов и поместить ее в очередь.
4. Сформировать очередь из 8 чисел. Найти сумму нечетных чисел и поместить в очередь.

6. Критерии оценивания результатов освоения дисциплины (модуля)

6.1. Оценочные средства и критерии оценивания для текущей аттестации

Виды текущего контроля, предусмотренные рабочей программой дисциплины:

- 1) устный опрос;
- 2) выполнение практических заданий лабораторной работы.

1. Требования к устному ответу на вопросы к лабораторному занятию

Ответы студенты должны иллюстрировать конкретными примерами, опираться на теоретическую базу, прослеживать связи между теоретическими и практическими положениями учебной дисциплины, применять теоретические знания к решению вопросов.

Устный ответ предполагает грамотность устной речи; убедительность устной речи; ясность, точность; строгая последовательность, иллюстрация.

Критерии оценки устного ответа: полнота и правильность ответа; логика изложения; степень осознанности и понимания изученного; связь теории с практикой.

«Зачтено» ставится, если студент:	- обстоятельно и достаточно полно излагает материал, возможны единичные ошибки; - обнаруживает понимание материала, может обосновать свои суждения, привести примеры; - строит ответ последовательно, возможны отдельные погрешности.
«Не зачтено» ставится, если студент:	- обнаружил незнание большей части темы (раздела, вопроса); - при ответе на вопрос искажает его смысл; - излагает материал беспорядочно и неуверенно.

Оценка может быть поставлена студенту как за единовременный ответ, так и за ответ, рассредоточенный во времени, т.е. за сумму ответов, данных в процессе занятий.

2. Требования к выполнению практических заданий лабораторной работы

Практическое задание лабораторной работы выполняется в письменном (печатном) виде. Это вид учебной работы студента по аналитической обработке информации, принятию самостоятельных решений, инициированию творческих идей.

Примеры практических заданий лабораторной работы

Лабораторная работа. Алгоритмы поиска

Задания для решения

1. Задача о парламенте*. На острове Новой Демократии каждый из жителей организовал партию, которую сам и возглавил. Любой из жителей острова может состоять не только в своей партии, но и в других партиях. К всеобщему удивлению, даже в самой малочисленной партии оказалось не менее двух человек. К сожалению, финансовые трудности не позволили создать парламент, куда вошли бы, как предполагалось по Конституции острова, президенты всех партий. Посоветовавшись, островитяне решили, что будет достаточно, если в парламенте будет хотя бы один член каждой партии.

Помогите островитянам организовать такой, как можно более малочисленный парламент, в котором будут представлены все партии.

Исходные данные: каждая партия и ее президент имеют один и тот же порядковый номер от 1 до N ($4 < N < 150$). Вам даны списки всех N партий острова Новой Демократии. Выведите предлагаемый вами парламент в виде списка номеров ее членов.

Например, для четырех партий: Список членов парламента 2 состоит из одного жителя с номером 2.

2. Какое наименьшее число ферзей можно расставить на доске так, чтобы они держали под боем все ее свободные поля?

Модификация задачи. Найти расстановку ферзей, которая одновременно решает задачу для досок 9×9 , 10×10 и 11×11 .

Указание. Задачу можно решать как обычным перебором, так и, представив доску как граф, путем поиска минимального доминирующего множества вершин.

3. Расставить на доске $N \times N$ ($N < 12$) N ферзей так, чтобы наибольшее число ее полей оказалось вне боя ферзей.

4. Расставить на доске как можно больше ферзей так, чтобы при снятии любого из них появлялось ровно одно неатакованное поле.

Показатели и критерии оценки задания:

полнота выполнения задания – от 0 до 3 баллов;

правильность выполнения задания (технологически) – от 0 до 3 баллов;

точность расчётов / логичность рассуждений – от 0 до 3 баллов;
аккуратность выполнения – от 0 до 3 баллов.

Шкала оценки:

- 0 – требование не выполнено;
- 1 – требование выполнено частично;
- 2 – требование выполнено, но есть недочёты;
- 3 – требование выполнено.

«зачтено» – 9 баллов и более;

«не зачтено» – менее 9 баллов.

Для получения оценки «зачтено» по выполнению практических заданий лабораторной работы студент должен получить оценку зачтено по каждому выполнению практического задания лабораторной работы из п.5 данной программы.

6.2. Оценочные средства и критерии оценивания для промежуточной аттестации

Критерий получения зачета - 6,7,8 семестр

Зачет выставляется по результатам работы студента в течение семестра.

Для получения зачета студент должен:

- уметь отвечать на теоретические вопросы, рассмотренные на лекциях;
- уметь решать задачи, предложенные на лабораторных занятиях.

Шкала оценивания навыков для получения зачета:

Количество лабораторных работ за которые получено «зачтено»	Оценка
15-18	«Зачтено»
Менее 15	«Не зачтено»

9 семестр

Вопросы к экзамену

1. Среда MIT App inventor. Интерфейс пользователя.
2. Среда MIT App inventor. Загрузка и установка приложения на устройство.
3. Среда MIT App inventor. Загрузка .apk файла на мобильное устройство.
4. Среда MIT App inventor. Компоненты игрового приложения. Разрешение экрана.
5. Практические приемы создания игровых приложений. Кнопки. Списки.
6. Практические приемы создания игровых приложений. Игровые приложения с несколькими экранами.
7. Практические приемы создания игровых приложений. Обмен данными между экранами.
8. Совместная разработка игровых приложений.
9. Особенности разработки математических игровых приложений
10. Особенности разработки игрового приложения для определенной возрастной группы
11. Особенности разработки игровых образовательных приложений

Типовые задачи для подготовки к экзамену

Разработайте игровые приложения, использующие простейшие элементы управления в мобильных устройствах:

1. Приложение “Загадка”.
2. Приложение “Отгадайка”.
3. Приложение “Угадай стихотворение”.

Разработайте игровые приложения, использующие обмен данными между экранами:

1. Приложение “Сказочные перемещения”.
2. Приложение “Смена цвета”.

3. Разработайте многооконное приложение “Ребенок превращается во взрослого”.

Разработайте игровые приложения, использующее списки:

1. Простейший графический редактор.

2. Приложение “Слайд-шоу”.

Разработайте графические приложения:

1. Приложение “Раскраска”.

2. Приложение “Пишем по образцу”.

Разработайте приложения с элементами анимации:

1. Приложение “Игра в шар”.

2. Приложение “Управляем движением объекта. Игра змейка”.

Разработайте игровые приложения, использующие медийные возможности MIT App inventor:

1. Приложение “Испорченный телефон”.

3. Приложение “Угадай слово”.

4. Приложение “Угадай фото”

Разработайте ГЕО-приложения:

1. Приложение “Где я?”.

2. Приложение “Угадай координаты” на основе компаса.

Разработайте математические игровые приложения:

1. Приложение “Конвертер систем счисления”

2.. Приложение “Тренажер” по произвольной теме.

Разработайте серию игровых приложений для определенной возрастной группы:

Первый период детства — от 3 до 7 лет

Второй период детства — от 8 до 12 лет

Подростковый возраст — от 13 до 16 лет

Юношеский возраст — от 17 до 21 года

Средний возраст

первый период — от 22 до 35 года

второй период — от 36 до 60 года

Пожилые люди — от 61 до 75 года

Разработайте серию игровых приложений по одному из разделов курса информатики в школе:

1. Информация и информационные процессы

2. Представление информации

3. Архитектура компьютера

4. Формализация и моделирование

5. Алгоритмизация и программирование

6. Информационные технологии

7. Социальная информатика

Образец экзаменационного билета

1. Среда MIT App inventor. Интерфейс пользователя.

2. Практические приемы создания игровых приложений. Обмен данными между экранами.

3. Задача.

Критерии выставления экзаменационной оценки

«Отлично» – студент полно и аргументировано отвечает по содержанию темы; может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из лекции, но и самостоятельно составленные; излагает материал последовательно и корректно, задача решена в полном объеме, алгоритмические и вычислительные ошибки отсутствуют, проведен анализ полученного решения.

«Хорошо» – студент полно и аргументировано отвечает по содержанию темы; может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из лекции, но и самостоятельно составленные; излагает материал последовательно и корректно, но допускает не более 1-2 ошибок, которые сам же исправляет, задача решена в полном объеме с незначительными техническими ошибками или отсутствует анализ результатов решения.

«Удовлетворительно» – студент излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки, задача решена не полностью или в решении присутствуют ошибки алгоритмического характера, незначительно влияющие на ход решения.

«Неудовлетворительно» – студент не знает ответ на вопрос, допускает существенные ошибки в формулировке определений и алгоритмов, искажающие их смысл, беспорядочно и неуверенно излагает материал, задача не решена или в решении присутствует значительное количество ошибок алгоритмического характера, существенно влияющих на ход решения.

7. Перечень основной и дополнительной учебной литературы

7.1. Основная литература

1. Гаврилов М.В. Информатика и информационные технологии : учебник для вузов / М. В. Гаврилов, В. А. Климов. — 4-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 383 с. — (Высшее образование). — ISBN 978-5-534-00814-2. — URL: <https://urait.ru/bcode/468473>.

2. Гниденко И. Г. Технологии и методы программирования : учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2025. — 235 с. — (Высшее образование). — ISBN 978-5-534-02816-4. — URL: <https://urait.ru/bcode/469759>.

3. Демин А. Ю. Информатика. Лабораторный практикум : учебное пособие для вузов / А. Ю. Демин, В. А. Дорофеев. — Москва : Издательство Юрайт, 2024. — 131 с. — (Высшее образование). — ISBN 978-5-534-08366-8. — URL : <https://urait.ru/bcode/470187>.

4. Зимин В. П. Информатика. Лабораторный практикум в 2 ч. Часть 1 : учебное пособие для вузов / В. П. Зимин. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2024. — 124 с. — (Высшее образование). — ISBN 978-5-534-11588-8. — URL: <https://urait.ru/bcode/470245>.

5. Зимин В.П. Информатика. Лабораторный практикум в 2 ч. Часть 2 : учебник для вузов / В. П. Зимин. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 153 с. — (Высшее образование). — ISBN 978-5-534-11590-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/563423>.

6. Зыков С. В. Объектно-ориентированное программирование: учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 151 с. — (Высшее образование). — ISBN 978-5-534-16941-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561434>.

7.2 Дополнительная литература

1. Зыков С. В. Программирование. Функциональный подход : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2021. — 164 с. — (Высшее образование). — ISBN 978-5-534-00844-9. — URL : <https://urait.ru/bcode/470387>.

2. Казанский, А. А. Программирование на Visual C# : учебное пособие для вузов / А. А. Казанский. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 192 с. — (Высшее образование). — ISBN 978-5-534-12338-8. — URL : <https://urait.ru/bcode/451467>.

3. Могилев А. В. Практикум по информатике: Учеб. пособие для студ. высш. учеб. заведений / А.В.Могилев, Н.И.Пак, Е.К.Хеннер; Под ред. Е.К.Хеннера. – 2-е изд., стер. – М.: Издательский центр «Академия», 2010. – 607 с.

4. Могилев А.В., Пак Н.И., Хеннер Е.К. Информатика. Учебное пособие для студентов пед. Вузов. М.: Издательский центр «Академия», 2012. – 848 с..

5. Трофимов В. В. Алгоритмизация и программирование: учебник для академического бакалавриата / В. В. Трофимов, Т. А. Павловская; под ред. В. В. Трофимова. – М.: Издательство Юрайт, 2018. – 137 с..

6. Тузовский, А. Ф. Объектно-ориентированное программирование : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2021. — 206 с. — (Высшее образование). — ISBN 978-5-534-00849-4. — URL : <https://urait.ru/bcode/470223>.

7. Черпаков И.В. Основы программирования : учебник и практикум для вузов / И. В. Черпаков. — Москва : Издательство Юрайт, 2021. — 219 с. — (Высшее образование). — ISBN 978-5-9916-9983-9. — URL : <https://urait.ru/bcode/469570>.

7.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Сайт Moodle СмолГУ. URL: www.cdo.smolgu.ru
2. Электронно-библиотечная система "Юрайт": <http://urait.ru>
3. Национальный открытый университет <http://intuit.ru>
4. Национальная платформа открытого образования <http://opened.ru>

8. Материально-техническое обеспечение

Учебная аудитория для проведения занятий лекционного типа, оснащенная стандартной учебной мебелью, мультимедиапроектором, ноутбуком, колонками и интерактивной доской.

Учебная аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная стандартной учебной мебелью, компьютерами с доступом к сети «Интернет» и ЭИОС СмолГУ.

Помещение для самостоятельной работы – компьютерный класс с доступом к сети «Интернет» и ЭИОС СмолГУ.

9. Программное обеспечение

Microsoft Open License (Windows XP, 7, 8, 10, Server, Office 2003-2016), Лицензия 66920993 от 24.05.2016, ежегодное обновление.

Microsoft Open License (Windows XP, 7, 8, 10, Server, Office 2003-2016), Лицензия 66975477 от 03.06.2016, ежегодное обновление.

Kaspersky Endpoint Security Лицензия 1FB6151216081242, ежегодное обновление.

Электронные библиотечные системы и электронная информационно-образовательная среда

Электронная библиотечная система «ЭБС ЮРАЙТ», Договор № 3074 от 15.11.2017, ежегодное обновление.

СДО Русский Moodle 3KL Norm с техническим обслуживанием, Акт на передачу прав №УТДЮ0001785 от 06.12.2016, ежегодное обновление.

Веб-браузер Google Chrome.

Система программирования MS Visual Studio 19 (язык программирования C#).